UNIFIED GENERATIVE VIDEO COMPRESSION WITH AUTOREGRESSIVE AND BIDIRECTIONAL VIDEO DIFFUSION

Anonymous authors
Paper under double-blind review

ABSTRACT

Video diffusion models can reconstruct visually compelling content from severely compressed representations at ultra-low bitrates, but they impose a rigid deployment trade-off. Bidirectional (BI) models achieve high quality through dense full-sequence attention, yet require the entire sequence before reconstruction and incur substantial latency. Autoregressive (AR) models enable low-delay streaming, but their restricted temporal context reduces compression efficiency. This tension is particularly relevant when a video stream must be decoded in real time yet is also stored for later, quality-oriented playback, making AR and BI models natural choices for the live and replay stages, respectively. To support both stages, we present **UniGVC**, a **Uni**fied **G**enerative **V**ideo **C**odec that supports two decoding modes from a single bitstream. UniGVC-BI employs a BI-DiT with dense attention for quality-oriented random-access decoding, whereas UniGVC-AR uses windowed AR-DiT for low-delay streaming. Both paths share a hierarchical latent codec that recurrently produces the same conditions with ultra-low bitrates, allowing UniGVC-BI to serve as a conditional teacher for single-step distillation of UniGVC-AR. A tiny conditional video decoder further reduces streaming latency while preserving most of the reconstruction quality. On standard 1080p benchmarks, UniGVC-BI achieves the best overall rate-distortion-perception trade-off among the evaluated methods and produces temporally stable video at rates as low as 0.0005 bits per pixel. Meanwhile, UniGVC-AR retains competitive reconstruction quality while decoding 1080p video at 36.85 FPS, over $250\times$ faster than UniGVC-BI. Code and models will be released.

1 INTRODUCTION

Video compression expects to preserve convincing visual content at rates where accurate transmission of pixels is impossible. Traditional standards such as H.265/HEVC and H.266/VVC (Sullivan et al., 2012; Bross et al., 2021), as well as distortion-optimized neural video codecs (Lu et al., 2019; Li et al., 2021; 2024; Jia et al., 2025), are highly effective at removing spatial and temporal redundancy. At ultra-low bitrates, however, minimizing pixel distortion encourages averaging over details, producing blur and erasing fine structure. Generative codecs (Qi et al., 2025; Guo et al., 2025; Mao et al., 2026; Ling et al., 2026) instead accept synthesized but visually consistent details and target the rate-distortion-perception trade-off (Blau & Michaeli, 2019).

Video diffusion transformer (DiT) can jointly reason about appearance and motion and is consequently well suited to coherent perceptual reconstruction (Peebles & Xie, 2023; Wan et al., 2025). Nevertheless, its advantages come at a substantial systems cost: dense bidirectional attention requires the whole coded segment, iterative denoising invokes the large model many times, and the accompanying 3D VAE decoder can itself dominate latency. Such a diffusion decoder is attractive for quality-oriented, delay-tolerant operation, but unsuitable for streaming scenarios.

Autoregressive video diffusion removes future evidence, while causal attention permits a frame or latent chunk to be emitted as soon as its condition arrives, and cached keys and values avoid recomputing the past (Yin et al., 2025; Huang et al., 2026b). This distinction is important in applications such as video conference and online teaching, where the same stream must be viewed in real time

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.

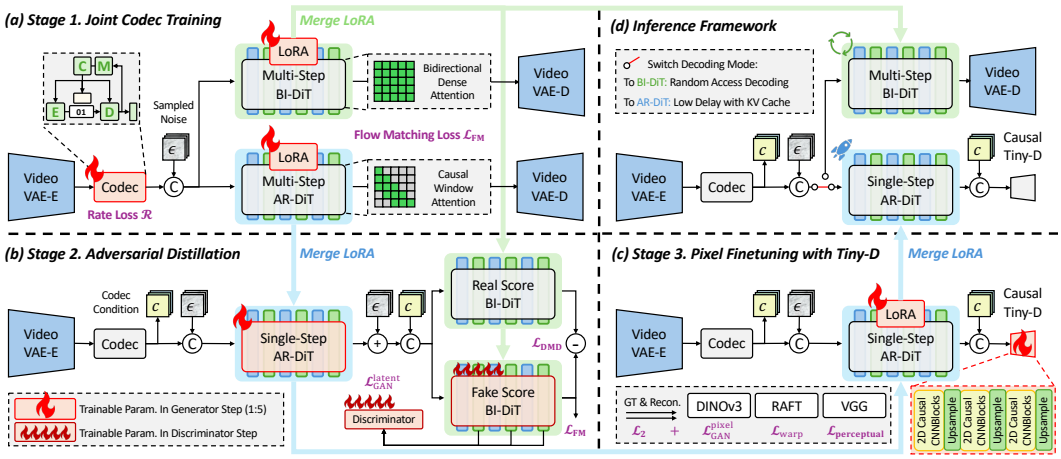

Figure 1: **Training and inference pipeline of UniGVC. (a)** Training stage 1 jointly adapts a shared latent codec and LoRA modules for a dense bidirectional DiT (BI-DiT) and a windowed causal DiT (AR-DiT). **(b)** Stage 2 merges the LoRAs and distills AR-DiT to one step using BI-DiT as the real score, fake score, and adversarial supervisor under the same codec condition. **(c)** Stage 3 refines AR-DiT and a conditional tiny decoder in pixel space. **(d)** At inference, the same codec condition selects either BI-DiT for random-access decoding or one-step AR-DiT for low-delay streaming.

and retained for later playback. A causal AR decoder can serve the live session with low delay. Once stored, a BI decoder can exploit both past and future context for higher-quality replay. However, existing generative video codecs typically bind the bitstream to one synthesis path.

We explore whether one compressed representation can instead support both regimes. Unlike video generation, the diffusion now receives a strong, temporally aligned codec signal reconstructed from the bitstream. This signal can serve as the autoregressive context. Conditioning on codec features rather than previously reconstructed pixels or clean latents permits parallel training with a shared diffusion timestep while retaining causal, cached inference. The same condition remains compatible with a bidirectional generator, allowing both diffusions to optimize jointly with one shared codec.

Based on this observation, we introduce **UniGVC** (Fig. 1), a unified generative video compression framework at ultra-low bitrates supporting bidirectional and autoregressive diffusion decoding in a continuous video latent space (Wan et al., 2025). We facilitate a recurrent hierarchical codec to compress video latents with ultra-low bitrates. The bitstream reconstructs a sequence of shared codec conditions. UniGVC-BI processes those conditions with dense bidirectional attention and iterative denoising, maximizing perceptual quality and temporal consistency for buffered random-access decoding. UniGVC-AR instead uses contextual autoregression and a windowed KV cache for low-delay causal streaming. We distill it into one step with the bidirectional teacher through distribution matching and adversarial supervision, then replace the costly 3D video VAE decoder with a condition-aware tiny causal decoder. Inference consequently supports switching between generative decoders for different decoding quality and latency with the same bits.

We summarize our contributions as:

- We present a unified generative video compression framework whose single bitstream supports both a quality-oriented bidirectional decoding mode and a real-time autoregressive decoding mode, covering random-access and low-delay use cases at ultra-low bitrates.
- We develop a hierarchical latent video codec for efficient bits allocation at severe compression. To enable autoregressive streaming, we distill AR-DiT into one step exploiting BI-DiT as the distribution teacher under the same codec condition. A conditional tiny decoder further breaks the real-time decoding bottleneck while maintaining most of the quality.
- Extensive experiments on video benchmarks demonstrate that UniGVC-BI achieves superior rate-distortion-perception trade-off and temporal stability at bitrates as low as 0.0005 bpp, while UniGVC-AR remains competitive and reaches 36.85 decoding FPS at 1080p.

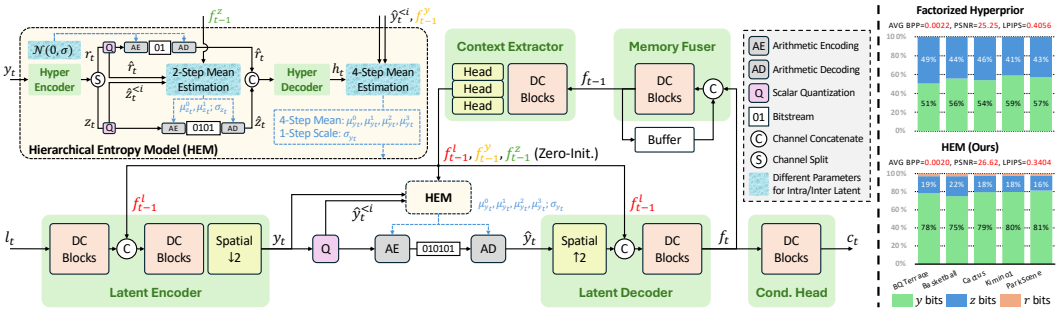

Figure 2: **Codec overview and analysis on bits allocation. Left:** The unified latent video codec compresses video latent l_t with hierarchical entropy modeling, updates recurrent feature memories, then synthesizes the shared condition c_t for later autoregressive or bidirectional video generation. Most modules are shared across intra and inter latent frames. "DC Blocks" stands for depthwise convolution blocks (Li et al., 2026). **Right:** Bits allocation comparison on HEVC-B. "Factorized Hyperprior" omits the z_t-r_t decomposition and applies the standard factorized entropy bottleneck, producing z_t with substantial bits without context modeling. In contrast, HEM assigns most bits to the context-modeled y_t and z_t, leaving less than 3% bits on weakly modeled r_t with improved compression efficiency and reconstruction quality.

2 Unified Generative Video Compression

2.1 Hierarchical Latent Video Codec for Ultra-Low Bitrates

UniGVC facilitates the temporal compression of the video VAE (Wan et al., 2025). Given a video $x \in \mathbb{R}^{T\times H\times W}$, the VAE encoder maps it to a continuous latent sequence l with 16 channels, $8\times$ spatial downsampling, and $4\times$ temporal downsampling (with a separately represented first frame), making latent coding substantially cheaper than pixel-domain coding. As overviewed in Fig. 2, we construct a recurrent codec unifying intra and inter latent coding to meet the autoregressive streaming. A hierarchical entropy model is introduced to refine bits allocation under ultra-low bitrates.

Unified Recurrent Latent Coding. For latent frame l_t, the latent encoder produces a primary latent y_t using the previous decoded feature f^l_{t-1}. The latent decoder reconstructs a feature f_t, updates a recurrent memory buffer (Guo et al., 2025), and projects it to the condition c_t. For the first latent frame, temporal buffer and inputs are zero initialized and a small set of intra-specific entropy layers is activated. This unified intra/inter design keeps every c_t in the same feature space and makes joint optimization with both diffusion decoders possible.

Hierarchical Entropy Model. At extreme rates, conventional factorized hyperprior (Ballé et al., 2018) typically consumes a considerable part of the bit budget (Fig. 2 Right). We therefore decompose a residual r_t from the hyper latent z_t that is quantized then entropy-coded with a zero-mean Gaussian and learned channel-wise scales. Conditioned on $\hat{r}_t$ and temporal feature f^z_{t-1}, $\hat{z}_t$ is coded using a two-pass checkerboard partition (He et al., 2021). The hyper decoder reconstructs the hyperprior from $[\hat{z}_t; \hat{r}_t]$. With temporal feature f^y_{t-1}, $\hat{y}_t$ is modeled by a four-pass quadtree partition (Li et al., 2023). The resulting hierarchical entropy factorization and the expected bitrate are

$$p(\hat{y}_t, \hat{z}_t, \hat{r}_t \mid f^y_{t-1}, f^z_{t-1}) = p(\hat{r}_t) \prod_{i=1}^{2} p\left(\hat{z}^i_t \mid \hat{z}^{<i}_t, \hat{r}_t, f^z_{t-1}\right) \prod_{i=1}^{4} p\left(\hat{y}^i_t \mid \hat{y}^{<i}_t, \hat{h}_t, f^y_{t-1}\right), \tag{1}$$

$$\mathcal{R} = \frac{1}{THW} \sum_t \mathbb{E}[-\log_2 p(\hat{r}_t) - \log_2 p(\hat{z}_t \mid \cdot) - \log_2 p(\hat{y}_t \mid \cdot)]\,. \tag{2}$$

The r branch can now shrink toward zeros, allowing more bits to y and z that are modeled by stronger spatial and temporal context under different feature resolutions. Following DCVC-UF (Li et al., 2026), we predict the scales of all partitions in single evaluation and refine only their means across multiple checkerboard or quadtree passes. Arithmetic decoding therefore interacts with accelerators once at each of the r_t, z_t, and y_t levels, while progressive mean estimation remains on the device.

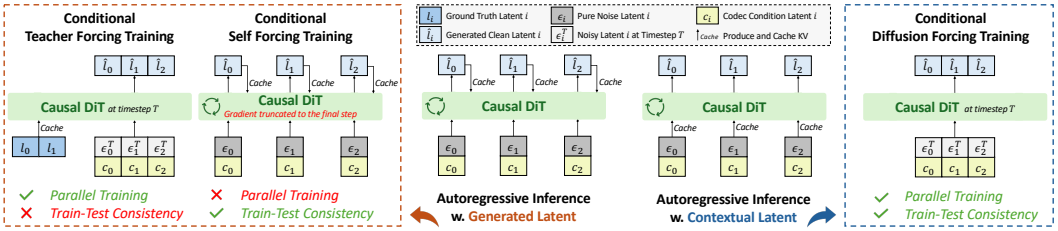

Figure 3: **Codec-conditioned autoregressive diffusion training and inference.** Autoregression through generated latents requires either teacher forcing (TF), which creates exposure bias, or self forcing (SF), which serializes training. UniGVC instead autoregresses through decoded codec conditions. Conditional diffusion forcing (DF) uses a shared diffusion timestep across frames, permits parallel training, and presents the same bitstream-derived condition at training and inference.

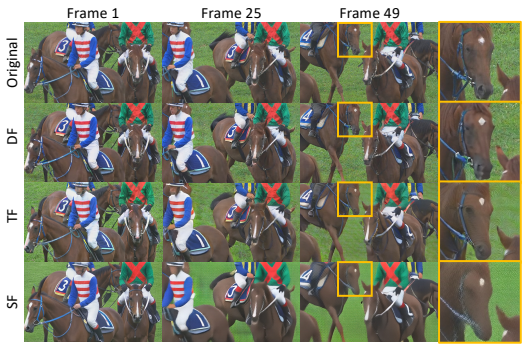

Figure 4: **Visual analysis of different codec-conditioned AR-DiT.** Videos are tested in 480p and 0.0045-0.005 bpp.



Figure 5: **Visual analysis on the codec condition and the conditional tiny decoder.** Videos are evaluated in 1080p and 0.0005 bpp.

Codec Initialization. Before incorporating diffusion, we pretrain the codec with

$$\mathcal{L}_{\text{codec}} = \lambda \cdot \mathcal{R} + \sum_t \omega_t \cdot \|l_t - c_t\|_2^2, \tag{3}$$

where ω_t implements hierarchical frame-quality weighting (Li et al., 2024; Guo et al., 2025). This stage supplies a stable, rate-efficient condition before the large generative models are attached.

2.2 Joint Training with Contextual Autoregressive Video Diffusion

Preliminaries: Autoregressive Video Generation. Autoregressive (AR) video diffusion predicts one latent frame (or chunk) at a time based on previous generation, involving three typical training paradigms. Teacher forcing (TF) (Gao et al., 2024) trains to denoise the next frame conditioned on a clean ground-truth prefix. It supports parallel training with a causal mask, but replaces that prefix with imperfect generated latents at inference, causing exposure bias. Diffusion forcing (DF) (Chen et al., 2024) assigns different noise levels to frames in a training sequence, leaving a similar train-test gap (Huang et al., 2026b; Zhu et al., 2026). Self forcing (SF) (Huang et al., 2026b) closes this gap by performing an AR rollout during training and conditioning on previously generated clean latents, but serializes training and typically truncates gradients through the rollout. These paradigms were adopted primarily for video generation, where generated latents are the only strong temporal observations. It remains unclear which AR paradigm is most effective when a learned codec condition is jointly optimized.

Autoregressive Paradigms with a Video Codec. As shown in Fig. 3, a codec permits two AR inference pipelines. *AR with generated clean latents* follows conventional AR video generation and predicts $\hat{l}_t$ from the current codec condition c_t and previously generated clean latents $\hat{l}_{<t}$. It corresponds to conditional TF or SF training. TF is parallel but train-test inconsistent, whereas SF is consistent but requires a sequential self-rollout, making joint optimization with a recurrent codec

expensive. The second, *AR with contextual latent*, is specific to video compression. Because the bitstream provides a frame-aligned condition at every time step, the model can instead use the causal condition sequence as history without feeding reconstructed latents back into later predictions. We train this path using a codec-conditioned variant of DF. Since $c_{\leq t}$ carries the temporal history rather than a differently noised latent prefix, all target latents can share a diffusion timestep and be optimized in parallel under causal attention. The conditions are decoded from the same bitstream during training and inference, so this formulation avoids conditioning on accumulated generation errors and allows every frame to provide gradients to the codec in parallel.

Contextual autoregression also follows established practice in neural video coding, where learned feature context is generally more effective for inter-frame prediction than a reconstructed frame alone (Li et al., 2021; Sheng et al., 2022; Li et al., 2023). An RGB frame has only three channels, and a clean latent in video VAE has only 16. By contrast, codec condition c_t supports much more learned channels (e.g., 256). It is not constrained to be directly reconstructible and can preserve diverse predictive cues about motion, structure, and color. Guided by these insights, we adopt contextual autoregression trained with conditional DF for UniGVC-AR. Fig. 4 validates this choice visually. At similar bitrates, DF better preserves appearance over time and exhibits less accumulated degradation than conditional TF and SF.

Attention and Joint Objective. Both diffusion branches are initialized from the Wan2.1-1.3B video DiT (Wan et al., 2025). BI-DiT preserves dense spatial-temporal attention. AR-DiT instead applies a causal temporal window of w latent frames. This masked attention admits streaming inference and bounds the KV-cache length. We use $w = 3$ based on the ablation in Sec. 4.3. LoRA adapters (Hu et al., 2021) are inserted into the two DiTs to preserve the generative priors. We jointly update the adapters and codec using flow matching loss $\mathcal{L}_{\mathrm{FM}}$. The objective for stage 1 is

$$\mathcal{L}_{\mathrm{Stage1}} = \lambda \cdot \mathcal{R} + \lambda_{\mathrm{FM}} \cdot \mathcal{L}_{\mathrm{FM}}. \tag{4}$$

Gradients from both decoding modes consequently shape the same bitstream representation. After training, the dense branch supports sequence-level generation from a buffered coded segment, whereas the causal branch processes newly available c_t with bounded KV memory.

3 Toward Real-Time Autoregressive Video Diffusion Decoder

3.1 Adversarial Distribution Matching Distillation

The multi-step AR-DiT is still expensive for real-time. We distill it into one step using the stronger BI-DiT as the distribution teacher and adversarial supervisor under the same codec condition c.

Conditional Distribution Matching Distillation. After merging the LoRAs, we freeze the codec and initialize the real-score network $\mathrm{s}_{\mathrm{real}}$ and the fake-score network $\mathrm{s}_{\mathrm{fake}}$ from BI-DiT, and optimize the full generator G_θ and $\mathrm{s}_{\mathrm{fake}}$. For flow time $\tau \in [0, 1]$ and noise $\epsilon \sim \mathcal{N}(0, I)$, we perturb the generated latent $\hat{l}$ into $\hat{l}_\tau = (1 - \tau) \cdot \hat{l} + \tau \cdot \epsilon$. Following DMD (Yin et al., 2024b;a), the gradient is estimated by the conditional score differences weighted by a normalizer $w(\tau)$:

$$\nabla_\theta \mathcal{L}_{\mathrm{DMD}} = \mathbb{E}\left[w(\tau) \big(\mathrm{s}_{\mathrm{fake}}(\hat{l}_\tau, \tau; c) - \mathrm{s}_{\mathrm{real}}(\hat{l}_\tau, \tau; c) \big) \frac{\partial \mathrm{G}}{\partial \theta} \right]. \tag{5}$$

Appending a Discriminator Head. Distribution matching alone aligns the student with the imperfect teacher distribution. We therefore add a latent adversarial objective (Lin et al., 2025) that directly distinguishes real and generated video latents. From intermediate features F_i of $\mathrm{s}_{\mathrm{fake}}(\hat{l}_\tau, \tau; c)$, a discriminator head D_ϕ with cross-attention, MLP, and learnable query q_i computes the scalar logit

$$a_i = \mathrm{MLP}_i(\text{Cross-Attn}(\mathrm{q}_i, \mathrm{k}_i(F_i), \mathrm{v}_i(F_i))), \qquad \mathrm{D}_\phi(\hat{l}_\tau, \tau; c) = \mathrm{MLP}([a_1; \ldots; a_n]). \tag{6}$$

With the non-saturating GAN loss (Goodfellow et al., 2014), the adversarial and stage objectives are

$$\mathcal{L}^{\mathrm{latent}}_{\mathrm{GAN}} = \mathbb{E}[\mathrm{softplus}(\mathrm{D}_\phi(l_\tau, \tau; c))] + \mathbb{E}\Big[\mathrm{softplus}(-\mathrm{D}_\phi(\hat{l}_\tau, \tau; c))\Big], \tag{7}$$

$$\mathcal{L}_{\mathrm{Stage2}} = \mathcal{L}_{\mathrm{DMD}} + \lambda^{\mathrm{latent}}_{\mathrm{GAN}} \cdot \mathcal{L}^{\mathrm{latent}}_{\mathrm{GAN}}. \tag{8}$$

During discriminator updates, adversarial gradients are stopped at the fake-score backbone, so the head D_ϕ is optimized by classification while $\mathrm{s}_{\mathrm{fake}}$ is updated only by its flow matching loss. We perform five D_ϕ and $\mathrm{s}_{\mathrm{fake}}$ updates per G_θ update. These separations stabilize adversarial learning while retaining an accurate estimate of the generator's evolving distribution.

3.2 Pixel Finetuning with Conditional Tiny Decoder

Once AR-DiT is reduced to one step, the original video VAE decoder becomes the dominant latency term because of its deep 3D causal convolutions and attention. We replace it in UniGVC-AR with a lightweight tiny decoder composed of 2D causal memory blocks and PixelShuffle upsampling (Shi et al., 2016; Huang et al., 2026a). Each block fuses its current-frame activation with a cached feature from the preceding frame, preserving causal temporal information without 3D convolution.

Pixel Finetuning. We optimize the entire tiny decoder together with newly inserted LoRA in the one-step AR-DiT. Given the reconstruction video $\hat{x}$. We construct a DINOv3 (Siméoni et al., 2025) feature discriminator to supply the pixel adversarial term $\mathcal{L}_{\mathrm{GAN}}^{\mathrm{pixel}}$, and improve temporal stability with flow warping error $\mathcal{L}_{\mathrm{warp}}$ (Lai et al., 2018) from a frozen RAFT (Teed & Deng, 2020). Perceptual supervision $\mathcal{L}_{\mathrm{p}}$ consists of LPIPS and DISTS measures from the VGG features (Zhang et al., 2018; Ding et al., 2020). The final pixel stage objective is

$$\mathcal{L}_{\mathrm{Stage3}} = \lambda_{\mathrm{MSE}} \cdot \|x - \hat{x}\|_2^2 + \lambda_{\mathrm{p}} \cdot \mathcal{L}_{\mathrm{p}}(x, \hat{x}) + \lambda_{\mathrm{GAN}}^{\mathrm{pixel}} \cdot \mathcal{L}_{\mathrm{GAN}}^{\mathrm{pixel}}(x, \hat{x}) + \lambda_{\mathrm{warp}} \cdot \mathcal{L}_{\mathrm{warp}}(\hat{x}). \quad (9)$$

Conditional Tiny Decoder. Inspired from the dual-branch decoding in image codec (Zhang et al., 2025; 2026), we make full use of the codec condition c to enhance the tiny decoder, whose first projection now receives the concatenation $[\hat{l}_t; c_t]$. In Fig. 5, we train the tiny decoder with condition only, generated latent only and their combination. The high-dimensional codec condition retains faithful geometry and color even at bitrates as low as 0.0005 bpp, and steadily improves the reconstruction quality when concatenated to condition the tiny decoder.

4 Experiments

4.1 Settings

Datasets. Our training corpus contains 100K 512×512 images from MLIC-100K (Jiang et al., 2025), 1M 720p clips sampled from Koala-36M (Wang et al., 2025), and 11K 1080p clips of at least 97 frames from UltraVideo (Xue et al., 2025). We resize and center-crop samples to the resolution required by each training stage. Evaluation uses the first 96 frames of the 1080p sequences in HEVC-B (Sullivan et al., 2012), UVG (Mercat et al., 2020), and MCL-JCV (Wang et al., 2016). Neural codecs receive RGB frames converted from YUV420 with BT.709.

Implementation. Codec initialization takes 100K steps on images, 300K steps on 29-frame clips, and 35K steps on 97-frame clips. The batch size is 8 and clip size is 256×320. Starting from $\lambda = 0.25$, we finetune ultra-low-rate operating points with $\lambda \in \{1, 2, 4, 8, 16\}$. Stage 1 takes 15K steps on 256×320 clips and 2K steps on 512×832 with a batch size of 32. Each DiT is then adapted independently for 600 steps on 1024×1920 clips using only flow matching. Stage 2 runs for 600 steps on 97-frame clips, while stage 3 takes 2.6K steps on 25-frame clips, both using 1024×1920 clips and a batch size of 32. All models are trained in BF16 on at most four NVIDIA B300 GPUs.

Evaluation. We compare with H.266/VVC using VTM-17.0 in low-delay (VTM-LD) and random-access (VTM-RA) configurations (Bross et al., 2021); distortion-oriented neural codecs DCVC-FM (Li et al., 2024) and DCVC-RT (Jia et al., 2025); and generative codecs GLC-Video (Qi et al., 2025), GLVC (Guo et al., 2025), GNVC-VD (Mao et al., 2026), GVC1D (Zheng et al., 2026), FreeGVC (Ling et al., 2026), and S2VC (Xue et al., 2026b). We measure distortion with PSNR and MS-SSIM (Wang et al., 2003), perceptual fidelity with LPIPS (Zhang et al., 2018), DISTS (Ding et al., 2020), FID (Heusel et al., 2017) and KID (Bińkowski et al., 2018). Temporal behavior is measured by flow warping error $\mathrm{E}_{\mathrm{warp}}^{*}$ ($\times 10^{-3}$) (Lai et al., 2018), and overall video quality by DOVER (Wu et al., 2023). UniGVC-BI performs 50-step iterative denoising while UniGVC-AR applies 1-step denoising. Latency is measured on a single B300 GPU.

4.2 Main Comparison

Rate-distortion-perception performance. Fig. 6 compares all methods over the ultra-low-rate region. UniGVC-BI forms the strongest overall frontier on all three datasets. It preserves structures while obtaining markedly better perceptual fidelity than MSE codecs at comparable rates, and it also improves the distributional FID and KID criteria over generative codecs. The gain is especially

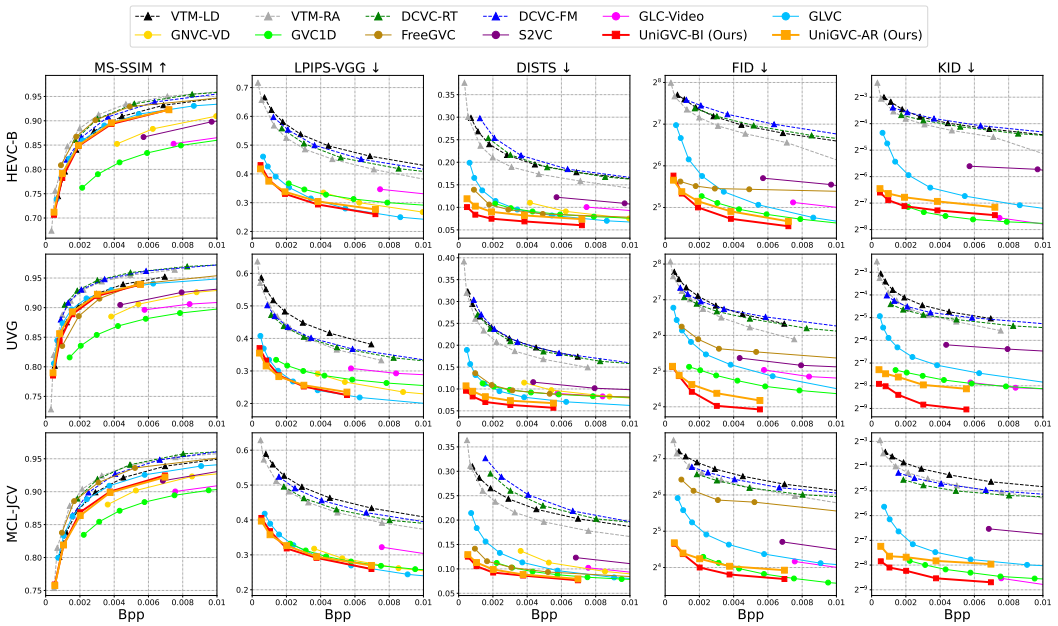

Figure 6: **Rate-distortion-perception curves on HEVC-B, UVG and MCL-JCV.** All video sequences are tested with the first 96 frames. UniGVC is evaluated with an intra-period of –1.

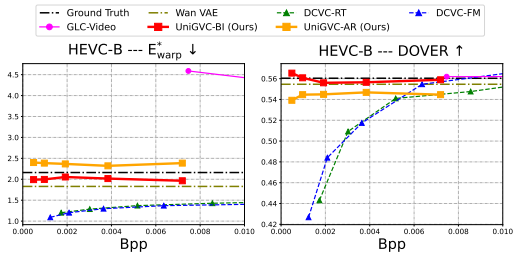

Figure 7: Temporal stability and video quality.

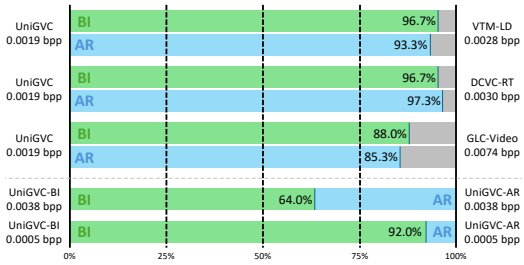

Figure 8: Human preference study on HEVC-B

pronounced near the lowest operating point of 0.0005 bpp. UniGVC-AR deliberately trades some of the bidirectional model's quality for latency, yet remains a competitive overall performance.

Temporal quality. As shown in Fig. 7, GLC-Video's frame-wise reconstruction incurs substantially larger warping error, illustrating that sharp individual frames do not guarantee a stable video. UniGVC-BI remains much closer to the temporally conservative distortion codecs while delivering higher DOVER quality. More importantly, its curve changes smoothly as the rate falls, including at 0.0005 bpp, rather than exhibiting a low-rate flicker collapse. UniGVC-AR is less stable than the bidirectional mode, but retains a competitive balance between temporal consistency and DOVER.

User Study. Fig. 8 records human preference of reconstructed HEVC-B videos under ultra-low bitrates. The study presents the ground truth video, and two reconstructions from different methods in random orders on the same high-resolution screen, collecting 30 votes from human observers for each comparison. Users express a clear preference for UniGVC over VTM-LD, DCVC-RT and GLC-Video, and an obvious tendency on UniGVC-BI against UniGVC-AR at lower bitrates.

Visual Comparison. Fig. 9 presents the quantitative results. Despite using substantially lower rates, UniGVC-BI recovers temporally stable texture and clean boundaries. UniGVC-AR preserves the same scene layout and motion with slight temporal artifacts, probably resulting from the tiny decoder where temporal growth adopts 2D convolutions. In contrast, DCVC-RT remains temporally steady but smooths thin edges and repeated texture, whereas GLC-Video synthesizes sharper content whose fine structures vary across frames and appear discontinuous in the temporal strips.

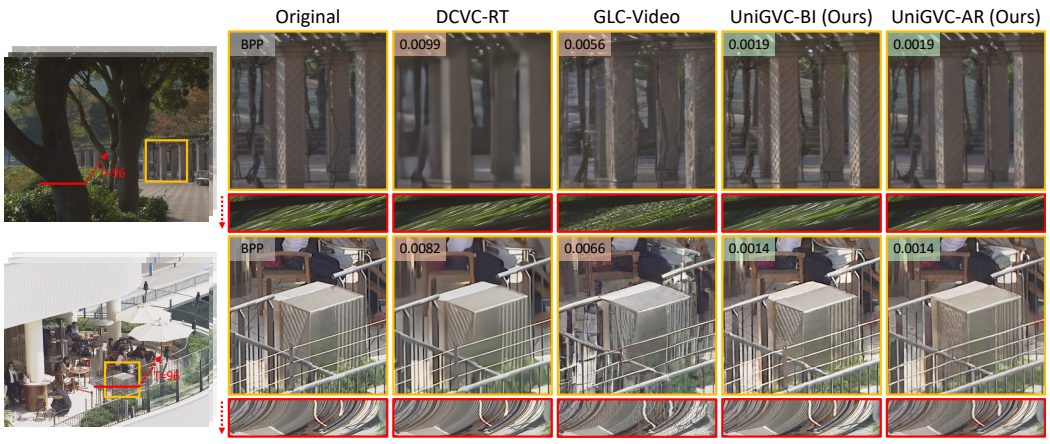

Figure 9: **Qualitative comparison on HEVC-B.** Each case displays the temporal profile (red box) at one row across 96 frames to observe temporal coherence. *Best viewed on screen for details.*

Table 2: **Ablation on decoding acceleration and enhancement.** We examine how architectural designs affect decoding speed, including 50-step bidirectional DiT (**BI-50**), 1-step autoregressive DiT (**AR-1**), original Wan VAE decoder (**Orig.**), and tiny decoder (**Tiny**). We also study how training strategies improve the perceptual fidelity, including adversarial distribution matching distillation (**Adv. DMD**), pixel finetuning (**Pixel FT.**), and conditional tiny decoder (**Cond. VAE-D**). Decoding speed and quality are evaluated on HEVC-B at 1080p, using four bitrates in 0.0005-0.004 bpp. Final choises for UniGVC-BI and UniGVC-AR are highlighted in color.

Exp.	Architecture		Training Strategy			Latency (ms / frame)			FPS	Bjøntegaard Delta	
	DiT	VAE-D	Adv. DMD	Pixel FT.	Cond. VAE-D	Codec	DiT	VAE-D		BD-LPIPS↓	BD-DISTS↓
1	BI-50	Orig.	-	-	-	7.42	6948.61	69.34	0.14	**0**	**0**
2	AR-1	Orig.	-	-	-	7.42	16.10	69.34	10.77	+0.069	+0.142
3	AR-1	Orig.	✓	-	-	7.42	16.10	69.34	10.77	+0.041	+0.045
4	AR-1	Tiny	✓	-	-	7.42	16.10	3.62	36.85	+0.074	+0.068
5	AR-1	Tiny	✓	✓	-	7.42	16.10	3.62	36.85	+0.058	+0.048
6	AR-1	Tiny	-	✓	✓	7.42	16.10	3.62	36.85	+0.075	+0.079
7	AR-1	Tiny	✓	✓	✓	**7.42**	**16.10**	**3.62**	**36.85**	+0.001	+0.017

4.3 Ablation Study

Bidirectional and causal attention. Table 1 examines how attention types affect the end-to-end performance. We train codec with the DiT using different attention. Training objectives include the rate loss and the flow matching loss. We evaluate BD-rate (Bjontegaard, 2001) with four operating points in 0.001-0.004 bpp. Dense bidirectional attention offers the best compression efficiency, but runs at only 7.09 FPS. Causal attention reaches 24.45 FPS, and bounding its history yields a further large speedup at the cost of a consistently performance degradation. A window size of three reaches 62.18 FPS. Further reduction saves little additional time but sharply degrades quality. We therefore set $w = 3$ for UniGVC-AR to strike a balance between speed and quality.

Table 1: **Ablation on DiT attention.** BD-rate is measured on HEVC-C with 50-step denoising. FPS is measured on 1080p videos with 1 step.

Attention	DiT FPS	BD-rate (↓ %)	
		LPIPS	DISTS
Bidirectional	7.09	**-29.13**	**-60.71**
Causal	24.45	0	0
Causal ($w = 5$)	44.92	+2.63	+2.57
Causal ($w = 3$)	62.18	+4.96	+5.35
Causal ($w = 2$)	66.37	+12.25	+17.33
Causal ($w = 1$)	**68.23**	+40.11	+42.63

From slow bidirectional to fast autoregressive decoding. Table 2 tracks the complete decoding acceleration and enhancement. Replacing BI-50 by AR-1 reduces DiT latency from 6948.61 to 16.10 ms per frame, but the direct one-step inference leaves substantial quality degradation. Adversarial DMD narrows these gaps, confirming that a bidirectional score teacher can improve a causal student. At that point the original VAE decoder takes 69.34 ms and dominates the pipeline. The tiny decoder reduces this component to 3.62 ms and raises end-to-end decoding throughput from 10.77 to 36.85 FPS, although a direct decoder replacement largely hurts quality. Pixel fine-tuning and codec conditioning recover most of this loss: the final UniGVC-AR lags behind UniGVC-BI by only 0.001 in BD-LPIPS and 0.017 in BD-DISTS on average. However, these small metric gaps between the two modes do not necessarily reflect the actual difference in perceived visual quality.

5 Related Work

5.1 Generative Video Compression

Early generative codecs introduced adversarial losses into image and video compression (Agustsson et al., 2019; Mentzer et al., 2020; Yang et al., 2022). Inspired by image codecs (Jia et al., 2024; Xue et al., 2026a; Zhang et al., 2025; 2026), recent video methods witnessed a growing trend on exploiting pretrained generative latent spaces or diffusion priors. GLC-Video compresses discrete generative latents (Qi et al., 2025), whereas GLVC uses a continuous video tokenizer and recurrent latent coding (Guo et al., 2025). DiffVC applies image-diffusion for perceptual enhancement (Ma & Chen, 2025). Concurrent designs explore compact one-dimensional latents (Zheng et al., 2026) or a single-step image diffusion decoder (Xue et al., 2026b). Video diffusion models display strong potential in alleviating temporal flickering. To this end, GNVC-VD performs sequence-level latent refinement with a video DiT (Mao et al., 2026); and Free-GVC codes trajectories of a pretrained video diffusion model without task-specific training (Ling et al., 2026). These approaches advance perceptual quality, but expose only one fixed decoding trade-off. UniGVC instead transmits a decoder-agnostic condition that can drive either a multi-step bidirectional generator or a one-step causal generator.

5.2 Autoregressive Video Diffusion and Acceleration

Standard video diffusions use dense bidirectional attention and iterative sampling, which provide strong sequence-level coherence but prevent online decoding. CausVid converts a bidirectional DiT into a few-step causal generator and applies asymmetric distribution-matching distillation (DMD) (Yin et al., 2025; 2024b;a). Self Forcing addresses exposure bias by training on autoregressively generated histories (Huang et al., 2026b), while Causal Forcing studies the architectural mismatch between bidirectional teachers and causal students (Zhu et al., 2026). Adversarial post-training provides another route to one-step video synthesis (Lin et al., 2025). These advances target open-ended generation, where past generated frames are the principal temporal condition and training must carefully reproduce their inference-time distribution.

6 Conclusion

This work presented UniGVC, a unified generative video compression framework that decouples the transmitted representation from the receiver's quality-latency choice for ultra-low-rate scenarios. A hierarchical latent codec produces one bitstream and one sequence of compression-aware conditions, driving either a multi-step bidirectional DiT for high-quality random-access decoding or a distilled one-step autoregressive DiT with a causal tiny decoder for low-delay streaming. Context-based autoregression enables parallel, train-test-consistent optimization, while adversarial distribution matching and pixel fine-tuning preserve much of the quality after acceleration. Across 1080p benchmarks, UniGVC-BI provides the strongest overall rate-distortion-perception trade-off among the evaluated methods, and UniGVC-AR reaches 36.85 FPS with only a small perceptual gap.

Limitations. Although UniGVC avoids storing or transmitting duplicate bitstreams, the retaining two decoding parameter sets increases the storage of decoder-side model. In addition, as with generative codecs, plausible detail may not reproduce the source exactly at extreme rates. Applications requiring scientific fidelity should avoid aggressive operating points or transmit additional residual information. Future work includes shared-weight multi-mode DiTs and region enhancement.

AI USE STATEMENT

In this work, we used generative AI tools solely to assist with polishing and improving the clarity and readability of the manuscript text. We have not used generative AI tools to generate the main research ideas, methodology, experimental designs, experimental results, figures, tables, or scientific conclusions. All AI-assisted text was carefully reviewed and revised by the authors to ensure its accuracy, consistency, and faithful representation of the underlying research. The research content, experimental methodology, results, figures, and tables were designed, conducted, and produced by the authors based on the actual experimental work. We take full responsibility for the final content of this work, including all text, claims, results, and artifacts produced with the aid of generative AI.

REFERENCES

Eirikur Agustsson, Michael Tschannen, Fabian Mentzer, Radu Timofte, and Luc Van Gool. Generative adversarial networks for extreme learned image compression. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 221–231, 2019.

Johannes Ballé, David Minnen, Saurabh Singh, Sung Jin Hwang, and Nick Johnston. Variational image compression with a scale hyperprior. *arXiv preprint arXiv:1802.01436*, 2018.

Mikołaj Bińkowski, Danica J Sutherland, Michael Arbel, and Arthur Gretton. Demystifying mmd gans. 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 *International Conference on Machine Learning*, pp. 675–685. PMLR, 2019.

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

Boyuan Chen, Diego Martí Monsó, Yilun Du, Max Simchowitz, Russ Tedrake, and Vincent Sitzmann. Diffusion forcing: Next-token prediction meets full-sequence diffusion. *Advances in Neural Information Processing Systems*, 37:24081–24125, 2024.

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, 2020.

Kaifeng Gao, Jiaxin Shi, Hanwang Zhang, Chunping Wang, Jun Xiao, and Long Chen. Ca2-vdm: Efficient autoregressive video diffusion model with causal generation and cache sharing. *arXiv preprint arXiv:2411.16375*, 2024.

Ian J Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. *Advances in neural information processing systems*, 27, 2014.

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

Dailan He, Yaoyan Zheng, Baocheng Sun, Yan Wang, and Hongwei Qin. Checkerboard context model for efficient learned image compression. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 14766–14775. IEEE, 2021.

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. volume 30, 2017.

Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022.

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. *arXiv preprint arXiv:2106.09685*, 2021.

Jie Huang, Zihao Fan, Xiaoxiao Ma, Jun-hao Zhuang, Yuming Li, Zeyue Xue, Siming Fu, Haoran Li, Mingchen Zhong, Guohui Zhang, et al. Ultra flash: Scaling real-time streaming video generation to high resolutions. *arXiv preprint arXiv:2606.09150*, 2026a.

Xun Huang, Zhengqi Li, Guande He, Mingyuan Zhou, and Eli Shechtman. Self forcing: Bridging the train-test gap in autoregressive video diffusion. *Advances in Neural Information Processing Systems*, 38:167283–167308, 2026b.

Zhaoyang Jia, Jiahao Li, Bin Li, Houqiang Li, and Yan Lu. Generative latent coding for ultra-low bitrate image compression. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 26088–26098. IEEE, 2024.

Zhaoyang Jia, Bin Li, Jiahao Li, Wenxuan Xie, Linfeng Qi, Houqiang Li, and Yan Lu. Towards practical real-time neural video compression. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 12543–12552. IEEE, 2025.

Wei Jiang, Jiayu Yang, Yongqi Zhai, Feng Gao, and Ronggang Wang. Mlic++: Linear complexity multi-reference entropy modeling for learned image compression. *ACM Transactions on Multimedia Computing, Communications and Applications*, 21(5):1–25, 2025.

Wei-Sheng Lai, Jia-Bin Huang, Oliver Wang, Eli Shechtman, Ersin Yumer, and Ming-Hsuan Yang. Learning blind video temporal consistency. In *European conference on computer vision*, pp. 179–195. Springer, 2018.

Jiahao Li, Bin Li, and Yan Lu. Deep contextual video compression. *Advances in Neural Information Processing Systems*, 34:18114–18125, 2021.

Jiahao Li, Bin Li, and Yan Lu. Neural video compression with diverse contexts. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 22616–22626. IEEE, 2023.

Jiahao Li, Bin Li, and Yan Lu. Neural video compression with feature modulation. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 26099–26108. IEEE, 2024.

Jiahao Li, Wenxuan Xie, Zhaoyang Jia, Bin Li, Zongyu Guo, Xiaoyi Zhang, and Yan Lu. Ultra-fast neural video compression. *arXiv preprint arXiv:2606.04410*, 2026.

Shanchuan Lin, Xin Xia, Yuxi Ren, Ceyuan Yang, Xuefeng Xiao, and Lu Jiang. Diffusion adversarial post-training for one-step video generation. *arXiv preprint arXiv:2501.08316*, 2025.

Xiaoyue Ling, Chuqin Zhou, Chunyi Li, Yunuo Chen, Yuan Tian, Guo Lu, and Wenjun Zhang. Free-gvc: Towards training-free extreme generative video compression with temporal coherence. *arXiv preprint arXiv:2602.09868*, 2026.

Guo Lu, Wanli Ouyang, Dong Xu, Xiaoyun Zhang, Chunlei Cai, and Zhiyong Gao. Dvc: An end-to-end deep video compression framework. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 10998–11007. IEEE, 2019.

Wenzhuo Ma and Zhenzhong Chen. Diffusion-based perceptual neural video compression with temporal diffusion information reuse. *ACM Transactions on Multimedia Computing, Communications and Applications*, 21(12):1–22, 2025.

Qi Mao, Hao Cheng, Tinghan Yang, Libiao Jin, and Siwei Ma. Generative neural video compression via video diffusion prior. pp. 43239–43248, 2026.

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.

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.

William Peebles and Saining Xie. Scalable diffusion models with transformers. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 4172–4182. IEEE, 2023.

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, 2025.

Xihua Sheng, Jiahao Li, Bin Li, Li Li, Dong Liu, and Yan Lu. Temporal context mining for learned video compression. *IEEE Transactions on Multimedia*, 25:7311–7322, 2022.

Wenzhe Shi, Jose Caballero, Ferenc Huszár, Johannes Totz, Andrew P Aitken, Rob Bishop, Daniel Rueckert, and Zehan Wang. Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1874–1883, 2016.

Oriane Siméoni, Huy V Vo, Maximilian Seitzer, Federico Baldassarre, Maxime Oquab, Cijo Jose, Vasil Khalidov, Marc Szafraniec, Seungeun Yi, Michaël Ramamonjisoa, et al. Dinov3. *arXiv preprint arXiv:2508.10104*, 2025.

Gary J Sullivan, Jens-Rainer Ohm, Woo-Jin Han, and Thomas Wiegand. Overview of the high efficiency video coding (hevc) standard. *IEEE Transactions on circuits and systems for video technology*, 22(12):1649–1668, 2012.

Zachary Teed and Jia Deng. Raft: Recurrent all-pairs field transforms for optical flow. In *European conference on computer vision*, pp. 402–419. Springer, 2020.

Team Wan, Ang Wang, Baole Ai, Bin Wen, Chaojie Mao, Chen-Wei Xie, Di Chen, Feiwu Yu, Haiming Zhao, Jianxiao Yang, et al. Wan: Open and advanced large-scale video generative models. *arXiv preprint arXiv:2503.20314*, 2025.

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. IEEE, 2016.

Qiuheng Wang, Yukai Shi, Jiarong Ou, Rui Chen, Ke Lin, Jiahao Wang, Boyuan Jiang, Haotian Yang, Mingwu Zheng, Xin Tao, et al. Koala-36m: A large-scale video dataset improving consistency between fine-grained conditions and video content. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 8428–8437. IEEE, 2025.

Zhou Wang, Eero P Simoncelli, and Alan C Bovik. Multiscale structural similarity for image quality assessment. In *The thrity-seventh asilomar conference on signals, systems & computers, 2003*, volume 2, pp. 1398–1402. Ieee, 2003.

Haoning Wu, Erli Zhang, Liang Liao, Chaofeng Chen, Jingwen Hou, Annan Wang, Wenxiu Sun, Qiong Yan, and Weisi Lin. Exploring video quality assessment on user generated contents from aesthetic and technical perspectives. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 20087–20097. IEEE, 2023.

Naifu Xue, Zhaoyang Jia, Jiahao Li, Bin Li, Yuan Zhang, and Yan Lu. One-step diffusion-based image compression with semantic distillation. *Advances in neural information processing systems*, 38:37108–37144, 2026a.

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*, pp. 9752–9761, 2026b.

Zhucun Xue, Jiangning Zhang, Teng Hu, Haoyang He, Yinan Chen, Yuxuan Cai, Yabiao Wang, Chengjie Wang, Yong Liu, Xiangtai Li, et al. Ultravideo: High-quality uhd video dataset with comprehensive captions. *arXiv preprint arXiv:2506.13691*, 2025.

Ren Yang, Radu Timofte, and Luc Van Gool. Perceptual learned video compression with recurrent conditional gan. In *IJCAI*, pp. 1537–1544, 2022.

Tianwei Yin, Michaël Gharbi, Taesung Park, Richard Zhang, Eli Shechtman, Fredo Durand, and William T Freeman. Improved distribution matching distillation for fast image synthesis. *Advances in neural information processing systems*, 37:47455–47487, 2024a.

Tianwei Yin, Michaël Gharbi, Richard Zhang, Eli Shechtman, Fredo Durand, William T Freeman, and Taesung Park. One-step diffusion with distribution matching distillation. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 6613–6623. IEEE, 2024b.

Tianwei Yin, Qiang Zhang, Richard Zhang, William T Freeman, Fredo Durand, Eli Shechtman, and Xun Huang. From slow bidirectional to fast autoregressive video diffusion models. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 22963–22974. IEEE, 2025.

Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *2018 IEEE/CVF conference on computer vision and pattern recognition*, pp. 586–595. IEEE, 2018.

Tianyu Zhang, Xin Luo, Li Li, and Dong Liu. Stablecodec: Taming one-step diffusion for extreme image compression. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 17379–17389. IEEE, 2025.

Tianyu Zhang, Dong Liu, and Chang Wen Chen. Ultra-low bitrate perceptual image compression with shallow encoder. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 12118–12128, 2026.

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. *arXiv preprint arXiv:2603.15302*, 2026.

Hongzhou Zhu, Min Zhao, Guande He, Hang Su, Chongxuan Li, and Jun Zhu. Causal forcing: Autoregressive diffusion distillation done right for high-quality real-time interactive video generation. *arXiv preprint arXiv:2602.02214*, 2026.

A TEST SETTINGS

Test data. The source sequences of HEVC-B (Sullivan et al., 2012), UVG (Mercat et al., 2020), and MCL-JCV (Wang et al., 2016) are stored in YUV 4:2:0 format. We use the first 96 frames of every sequence. Traditional codecs operate directly on the original YUV data. For neural codecs, we convert the source frames to RGB using the BT.709 conversion matrix before encoding. The decoded outputs of traditional codecs are converted to RGB with the same procedure, and all objective metrics are computed between RGB reference and reconstruction frames.

Traditional codec. We evaluate H.266/VVC (Bross et al., 2021) with the VTM-17.0 reference software. VTM-LD uses the Low-Delay-B configuration `encoder_lowdelay_vtm.cfg` with $\mathrm{QP} \in \{37, 40, 43, 46, 49, 52, 55\}$. VTM-RA uses `encoder_randomaccess_vtm.cfg` with $\mathrm{QP} \in \{36, 39, 42, 45, 48, 51, 57, 63\}$. Encoding and decoding are performed in the native YUV 4:2:0 format; only the decoded sequence used for metric computation is converted to RGB.

Neural codecs. For the distortion-oriented baselines DCVC-FM (Li et al., 2024) and DCVC-RT (Jia et al., 2025), and for GLC-Video (Qi et al., 2025), we run the official code and released checkpoints locally. All frames are processed in RGB. The code or checkpoints of GLVC (Guo et al., 2025), GNVC-VD (Mao et al., 2026), GVC1D (Zheng et al., 2026), Free-GVC (Ling et al., 2026), and S2VC (Xue et al., 2026b) were not publicly available when our evaluation was completed. We therefore use rate-distortion-perception points provided directly by the authors for GLVC, GVC1D, and S2VC, and the available results reported in the papers for GNVC-VD and Free-GVC.

Metrics. We evaluate PSNR, MS-SSIM (Wang et al., 2003), and DISTS (Ding et al., 2020) with PyIQA under its default settings. LPIPS (Zhang et al., 2018), FID (Heusel et al., 2017) and KID (Bińkowski et al., 2018) are implemented with TorchMetrics. Since benchmarks contain relatively few high-resolution frames, FID and KID are computed from overlapping 256×256 RGB patches extracted from the original and reconstructed sequences. We use the official implementation for the flow warping error $\mathrm{E}^*_{\mathrm{warp}}$ (Lai et al., 2018) and DOVER (Wu et al., 2023). Lower LPIPS, DISTS, FID, KID, and $\mathrm{E}^*_{\mathrm{warp}}$ are better, whereas higher PSNR, MS-SSIM, and DOVER are better.

B IMPLEMENTATION DETAILS OF UNIGVC

B.1 CODEC DESIGN

Table 3 summarizes the primary codec modules. Spatial resolutions are measured relative to the input video: the Wan VAE first maps a frame to $1/8$ spatial resolution, the primary latent y_t is represented at $1/16$, and the hierarchical hyperlatents (r_t, z_t) are represented at $1/64$. The primary latent contains 256 channels. The hyper-analysis output contains 128 channels and is split into a 16-channel residual branch r_t and a 112-channel context-modeled branch z_t.

The codec is built mainly from depthwise convolution (DC) blocks (Li et al., 2026). Each block contains a residual 3×3 depthwise convolution followed by two 1×1 pointwise projections and a weighted SiLU activation; a second residual feed-forward branch expands the feature width by four. Downsampling uses PixelUnshuffle followed by a 1×1 projection, while upsampling uses a 1×1 projection followed by PixelShuffle. The latent analysis and synthesis transforms share their main parameters between intra and inter latent frames. Only entropy parameters are specialized for the first intra latent frame, and the rest inter latent frames.

Table 3: **Architecture of the primary modules in the hierarchical latent video codec.** "Dimension" denotes the dominant internal feature width, and spatial resolution is relative to the input.

Module	Latent Encoder	Latent Decoder	Hyper Encoder	Hyper Decoder	y Entropy Model	z Entropy Model	Context Extractor	Memory Fuser	Condition Head
Dimension	256	256	128	128	512	256	512, 256	512	256
DC blocks	8	8	2	2	4	4	4, 4	4	4
Resolution	$\frac{1}{8} \to \frac{1}{16}$	$\frac{1}{16} \to \frac{1}{8}$	$\frac{1}{16} \to \frac{1}{64}$	$\frac{1}{64} \to \frac{1}{16}$	$\frac{1}{16}$	$\frac{1}{64}$	$\frac{1}{8}$	$\frac{1}{8}$	$\frac{1}{8}$

Table 4: Stage-wise training schedule.

Phase	Trainable modules	Frames	Crop size	Steps	Learning rate
Codec initialization	Codec	1	512×512	100K	10^{-4}
Codec initialization	Codec	29	256×320	300K	10^{-4}
Codec initialization	Codec	97	256×320	20K, 10K,5K	5×10^{-5} , 2×10^{-5},10^{-5}
Stage 1 warm-up	BI/AR-DiT LoRA	97	256×320	15K	10^{-4}
Stage 1 joint	Codec + BI/AR-DiT LoRA	97	256×320	5K	10^{-5}
Stage 1 joint	Codec + BI/AR-DiT LoRA	97	512×832	2K	10^{-5}
Stage 1 adaptation	BI-DiT LoRA or AR-DiT LoRA	97	1024×1920	600 each	10^{-4}
Stage 2	AR-DiT + Fake score + GAN	97	1024×1920	600	10^{-5}
Stage 3	AR-DiT LoRA + Tiny decoder + GAN	25	1024×1920	1K,1K, 500,100	10^{-4},5×10^{-5} , 2×10^{-5},10^{-5}

B.2 Training Procedure

Table 4 details the complete training curriculum.

Codec initialization. We first train the codec independently of the diffusion models. The curriculum uses $(1, 29, 97, 97, 97)$ frames for $(100\text{K}, 300\text{K}, 20\text{K}, 10\text{K}, 5\text{K})$ steps, with learning rates $(10^{-4}, 10^{-4}, 5\times10^{-5}, 2\times10^{-5}, 10^{-5})$. The image stage uses 512×512 crops; all video stages use 256×320 crops and batch size 8. We first obtain an initialization with $\lambda = 0.25$ in Eq. 3, then derive the ultra-low-rate models using $\lambda \in \{1, 2, 4, 8, 16\}$ and additional 97-frame training. The hierarchical weights (Guo et al., 2025) for the first four VAE latent frames are $(12.0, 1.2, 0.5, 0.9)$; subsequent frames alternate between 0.5 and 0.9.

Stage 1: joint codec–diffusion training. Each rate point resumes from its codec checkpoint. We insert rank-32 LoRA modules into BI-DiT and AR-DiT. For the first 15K steps on 97-frame 256×320 clips, the codec is frozen and only the LoRAs are optimized. We then jointly optimize the codec and both LoRAs for 5K steps at 256×320 and 2K steps at 512×832, using batch size 32. Independent LoRA adaptation uses learning rate 10^{-4}, while joint optimization uses 10^{-5}. We set $\lambda_{\text{FM}} = 0.35$, the flow-time shift to 5.0, and classifier-free condition dropout to 0.1. The rate weight of the first latent frame is multiplied by 0.1 during joint training to preserve a stronger causal anchor. Finally, with the codec fixed, we adapt the two LoRAs independently for 600 steps on 97-frame 1024×1920 clips using only flow matching. UniGVC-BI is fixed after this stage.

Stage 2: adversarial distribution matching. We merge the stage-1 AR-DiT LoRA into its backbone and optimize the full one-step generator, online fake-score model, and discriminator head for 600 steps using 97-frame 1024×1920 clips and batch size 32. The codec and real-score BI-DiT remain frozen. We use learning rate 10^{-5} and $\lambda_{\text{GAN}}^{\text{latent}} = 0.05$. The discriminator pools fake-score features after one-third, two-thirds, and all of the DiT depth. We update the discriminator head and fake-score model five times for every generator update; adversarial gradients are stopped at the fake-score backbone during discriminator optimization, while the fake-score backbone itself is trained with standard flow matching to model the distribution of the one-step generator.

Stage 3: pixel fine-tuning. Pixel training uses 25-frame 1024×1920 clips, batch size 32, and 2.6K steps. The four phases contain $(1000, 1000, 500, 100)$ steps with learning rates $(10^{-4}, 5\times10^{-5}, 2\times10^{-5}, 10^{-5})$. We train the entire tiny decoder and new rank-8 LoRA modules inserted into the one-step AR-DiT, while keeping the codec fixed. The DINOv3 discriminator input is resized to 448 to control memory consumption and follows the discriminator design of Zhang et al. (2026). In Eq. 9, $(\lambda_{\text{MSE}}, \lambda_{\text{p}}, \lambda_{\text{GAN}}^{\text{pixel}}, \lambda_{\text{warp}}) = (0.5, 2.0, 0.1, 1.0)$; LPIPS-VGG and DISTS contribute equally within the perceptual loss $\mathcal{L}_{\text{p}}$.

B.3 Inference with Different Decoding Modes

Both modes use exactly the same encoder, entropy-coded representation, textual side information, and recurrent codec decoder. After entropy decoding reconstructs the condition sequence, the re-

Table 5: **Module parameter counts in millions.** We compare UniGVC with generative codecs GLVC and GNVC-VD implemented on a similar Wan2.1-1.3B framework. GLVC and GNVC-VD use reported module parameters from their papers due to unavailable code and checkpoints.

Method	VAE Encoder	Codec	DiT modules	VAE Decoder	Total
Wan2.1-1.3B (Wan et al., 2025)	53.6	-	1419.0	72.6	1545.2
GLVC (Guo et al., 2025)	53.6	100.4	-	72.6	226.6
GNVC-VD (Mao et al., 2026)	53.6	53.1	2154.5	72.6	2333.8
UniGVC-BI	53.6	82.0	1419.0	72.6	1627.2
UniGVC-AR	(Shared)	(Shared)	1419.0	13.5	1568.1

Table 6: Encoding and decoding speed per frame at different resolutions on a B300 GPU.

	Decoding mode	1920×1080	1080×720	640×480
Encoding	-	86 ms	39 ms	21 ms
Decoding	UniGVC-BI	7025 ms	1502 ms	410 ms
Decoding	UniGVC-AR	27 ms	19 ms	12 ms

ceiver selects a generative path without transcoding the video bitstream. For UniGVC-BI, the conditions of the requested coding unit are buffered and processed jointly by BI-DiT with dense bidirectional attention. We perform 50 denoising steps with classifier-free guidance (CFG) (Ho & Salimans, 2022) scale 3.0, then map the generated latent sequence to RGB with the original Wan video VAE decoder. This mode targets quality-oriented random-access playback. UniGVC-AR instead applies a single causal AR-DiT evaluation per latent frame without CFG. It maintains a windowed KV cache containing the most recent latent frames and immediately decodes each generated latent with the conditional tiny decoder. The bounded cache makes DiT computation linear in sequence length. Because the Wan latent space uses $4\times$ temporal downsampling and the tiny decoder performs the corresponding causal upsampling, streaming has an intrinsic four-frame output latency.

C Complexity Analysis

C.1 Module Parameters

Table 5 compares module parameter counts with generative codecs that use the same Wan VAE or video-diffusion family. Wan2.1-1.3B (Wan et al., 2025) is included as a reference. GNVC-VD's DiT count includes its added control modules. The frozen text encoder is excluded from every row. We discuss caption generation and transmission separately in Sec. D. The tiny decoder in UniGVC-AR reduces pixel-decoder parameters from 72.6M to 13.5M, a $5.38\times$ reduction. The codec is shared between UniGVC-BI and UniGVC-AR, whereas their DiTs are separate mode-specific checkpoints. Thus, supporting both modes does not duplicate the transmitted representation, although a receiver that installs both modes must store both DiT parameter sets.

C.2 Practical Latency

Table 6 reports average conding speed on a B300 GPU. Encoding is shared by the two modes and includes the Wan VAE encoder, latent codec, and entropy encoding. Decoding includes entropy decoding, generative latent reconstruction, and pixel decoding. At 1920×1080, replacing 50-step BI-DiT and the original VAE decoder with one-step AR-DiT and the tiny decoder reduces latency from approximately 7025 to 27 ms per frame, a $250\times$ speedup. The unrounded measurement corresponds to the 36.85 FPS reported in the main paper. UniGVC-AR remains real-time at all tested resolutions, while UniGVC-BI deliberately spends substantially more computation for its quality-oriented operating point. The current 1080p encoder takes 86 ms per frame.

Table 7: **Statistics of the sequence-specific captions and complete prompts.** Averages are computed per sequence. The complete prompt includes the fixed 24-word general prompt, whereas the original byte count excludes it and corresponds to the original size in Table 8.

Dataset	No. of sequences	Sentences per caption	Sequence words	Full-prompt words	Original bytes
HEVC-B	5	4.8	84.8	108.8	517.8
UVG	7	4.7	112.4	136.4	684.9
MCL-JCV	30	5.3	124.3	148.3	759.6

Table 8: **Lossless compression of the transmitted sequence-specific captions.** Ratios are relative to the original UTF-8 string. Extra bpp amortizes one caption over 96 video frames at 1920×1080.

Dataset	Original bytes	Zlib bytes	Zstd+dict. bytes	Zlib ratio	Zstd ratio	Original extra bpp	Zlib extra bpp	Zstd+dict. extra bpp
HEVC-B	517.8	305.0	184.2	0.5890	0.3557	2.081×10^{-5}	1.226×10^{-5}	7.403×10^{-6}
UVG	684.9	391.6	233.4	0.5718	0.3408	2.752×10^{-5}	1.574×10^{-5}	9.380×10^{-6}
MCL-JCV	759.6	426.5	245.2	0.5614	0.3228	3.053×10^{-5}	1.714×10^{-5}	9.854×10^{-6}

D TEXTUAL PROMPT

D.1 VIDEO CAPTION

The pretrained Wan DiT is text-conditioned. At the encoder, we therefore use Qwen3.5-397B-A17B to generate one sequence-specific caption for each 96-frame test video. This caption is transmitted as a losslessly compressed UTF-8 string and recovered exactly at the decoder. Both UniGVC modes concatenate it with the same fixed general quality prompt. The frozen Wan T5 encoder then maps the complete prompt to at most 512 text tokens; only the sequence-specific text, rather than its embedding tensor, is signaled in the bitstream.

We use the following captioning instruction:

> "Watch the video carefully and write one detailed, factual paragraph of 80–150 words. When applicable, describe (i) the setting and location, (ii) the main subjects and their appearance, (iii) actions, motion, and event progression, (iv) camera movement and framing, and (v) lighting, color tone, and atmosphere. Do not add a preamble, bullet points, headings, or quotation marks. Be objective, specific, and dense in detail, and avoid vague filler."

This instruction encourages the sequence caption to describe information useful for conditional reconstruction rather than subjective interpretation. We use the following 24-word general quality prompt for every sequence:

> "Photorealistic video, realistic motion, natural environmental dynamics, physically plausible interactions, consistent temporal behavior, cinematic composition, documentary-style realism, high visual fidelity, fine-grained details, 4K quality."

Because this general prompt is fixed and shared by all videos, it is stored as part of the codec configuration and incurs no per-sequence rate. Table 7 summarizes the sequence-specific captions used in our evaluation. They contain 4.7-5.3 sentences and 84.8-124.3 words on average. After concatenating the fixed prompt, the complete prompts contain 108.8, 136.4, and 148.3 words on HEVC-B, UVG, and MCL-JCV, respectively. The transmitted sequence-caption strings average 517.8-759.6 bytes before lossless compression.

D.2 PROMPT COMPRESSION

We train a 512-KiB zstandard dictionary from one million non-empty captions sampled from the Koala-36M training corpus. The dictionary is fixed and distributed once with the model; it is not transmitted for each video. Each sequence-specific test caption is encoded with zstandard level 22

Table 9: **Effect of sequence-specific captions on HEVC-B.** BD-rate (%) is measured relative to the corresponding model using only the fixed general prompt. Lower is better; a negative value denotes a rate reduction at matched quality.

(a) UniGVC-BI

Prompt	LPIPS	DISTS	FID	KID
General only	0.00	0.00	0.00	0.00
+ sequence	-1.88	-11.19	-11.06	-17.11

(b) UniGVC-AR

Prompt	LPIPS	DISTS	FID	KID
General only	0.00	0.00	0.00	0.00
+ sequence	-2.64	-6.04	-14.29	-16.11



Original | General Caption Only 0.00052 bpp | General Caption + Sequence Caption 0.00052 bpp + 6.7×10^{-6} bpp

[Sequence Caption] A sunny urban park with a wide paved pathway running through lush greenery and mature trees, with a pergola extending into the distance. A cyclist wearing an orange shirt rides steadily away from the camera along the center of the path. Near the end of the shot, two distant cyclists appear and ride toward the camera from the opposite direction. The camera remains at eye level throughout the scene.
[General Caption] Photorealistic video, realistic motion, natural environmental dynamics, physically plausible interactions, consistent temporal behavior, cinematic composition, documentary-style realism, high visual fidelity, fine-grained details, 4K quality.

Figure 10: **Qualitative effect of the sequence-specific caption.** The columns show the source frame, UniGVC-BI's reconstruction using only the fixed general prompt, and reconstruction after adding the sequence caption. Both reconstructions use the same 0.00052-bpp bitstream; the sequence caption adds only 6.7×10^{-6} bpp. The exact sequence and general prompts are shown below the examples.

using this dictionary. We compare against Python zlib at level 9 and verify exact recovery after decompression for both methods. Table 8 reports the compressed sizes and compression ratios. To express textual side information on the same scale as the video rate, we amortize one sequence caption over a 96-frame 1920×1080 video. Dictionary-based zstandard reduces the average size by 39.6–42.5% relative to zlib and limits the additional rate to 7.40×10^{-6}–9.85×10^{-6} bpp. Even relative to our lowest visual operating point of 0.0005 bpp, this overhead is only 1.5–2.0%. The fixed general prompt is excluded from these values because it requires no per-video signaling.

D.3 Discussion on Prompt

To isolate the value of sequence-specific semantic side information, we compare the complete prompt against the fixed general quality prompt alone while keeping the codec and decoder unchanged. Table 9 reports BD-rate relative to the corresponding general-only baseline on HEVC-B. Adding the sequence caption improves all four perceptual criteria for both decoding modes. UniGVC-BI reduces BD-rate by 1.88% in LPIPS, 11.19% in DISTS, 11.06% in FID, and 17.11% in KID. UniGVC-AR yields corresponding reductions of 2.64%, 6.04%, 14.29%, and 16.11%. The consistent improvements for both decoders indicate that the caption complements the shared codec representation rather than benefiting only one synthesis path. The larger gains in FID and KID than in LPIPS further suggest that sequence-level text primarily helps resolve semantic and distributional ambiguity at these extremely low rates.

Fig. 10 visualize the influence. With only the fixed general prompt, the decoder is informed of the desired visual style but receives no textual description of this particular scene. The sequence caption supplies the setting, subjects, motion, and camera behavior that are difficult to communicate through a 0.00052-bpp visual code alone. Adding it produces a reconstruction that more faithfully reflects the source content, while its compressed rate is only 6.7×10^{-6} bpp for this example. Together

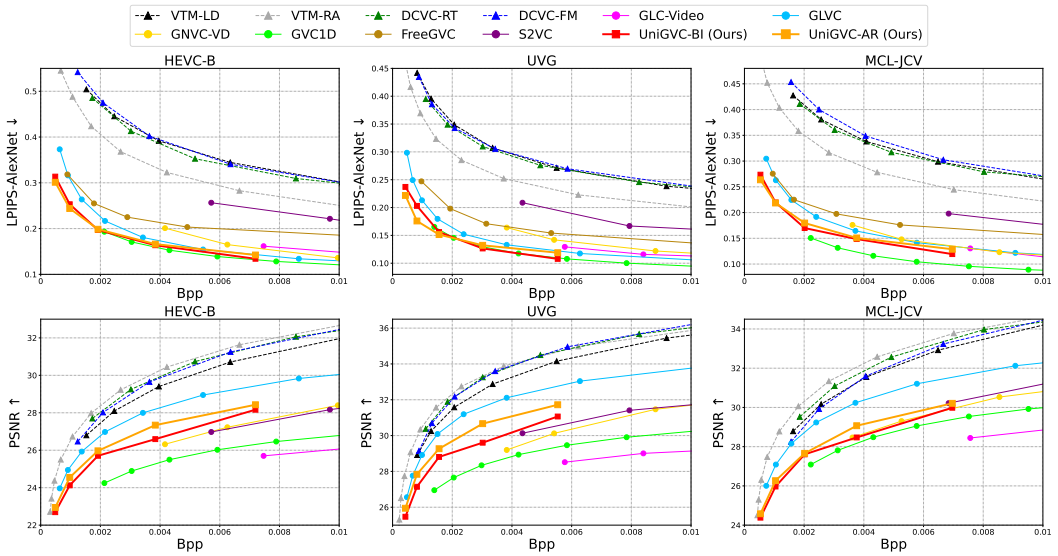

Figure 11: **Additional comparisons in terms of LPIPS-AlexNet and PSNR on HEVC-B, UVG, and MCL-JCV.** All methods follow the same 96-frame evaluation protocol as Fig. 6.

with the quantitative gains in Table 9, this example supports using a concise sequence description as low-cost semantic side information rather than relying on a generic quality prompt alone.

E Additional Quantitative Comparison

Fig. 11 complements the main comparison with LPIPS using an AlexNet backbone and with pixel-domain PSNR. The LPIPS-AlexNet results follow the perceptual trend observed with LPIPS-VGG in Fig. 6: UniGVC-BI and UniGVC-AR remain on or close to the favorable low-rate frontier across HEVC-B, UVG, and MCL-JCV. Pixel-level distortion metrics such as PSNR favor overly smooth or blurred reconstructions at ultra-low bitrates, which deviate substantially from human perception. We therefore report PSNR only as a supplementary distortion measure for reference, while evaluating ultra-low-rate compression primarily in terms of perceptual fidelity, temporal coherence, and human preference.

F Additional Qualitative Comparison

Fig. 12–15 provide additional qualitative comparisons on UVG and MCL-JCV. Each example presents reconstructed patches together with a 96-frame temporal profile, allowing spatial fidelity and temporal coherence to be assessed jointly. Both UniGVC-BI and UniGVC-AR deliver faithful, temporally consistent reconstructions while using fewer bits than GLC-Video and DCVC-RT.

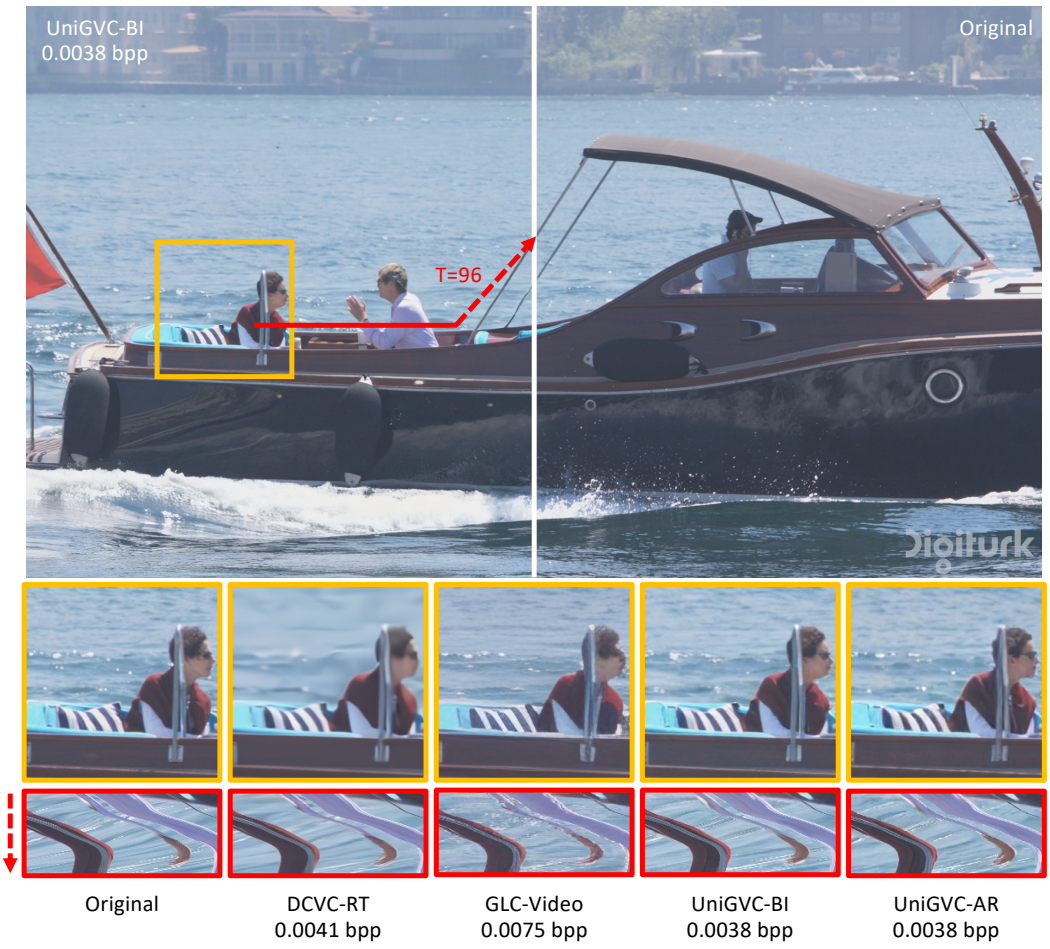

Figure 12: **Visual example from UVG.** *Best viewed on screen for details.*

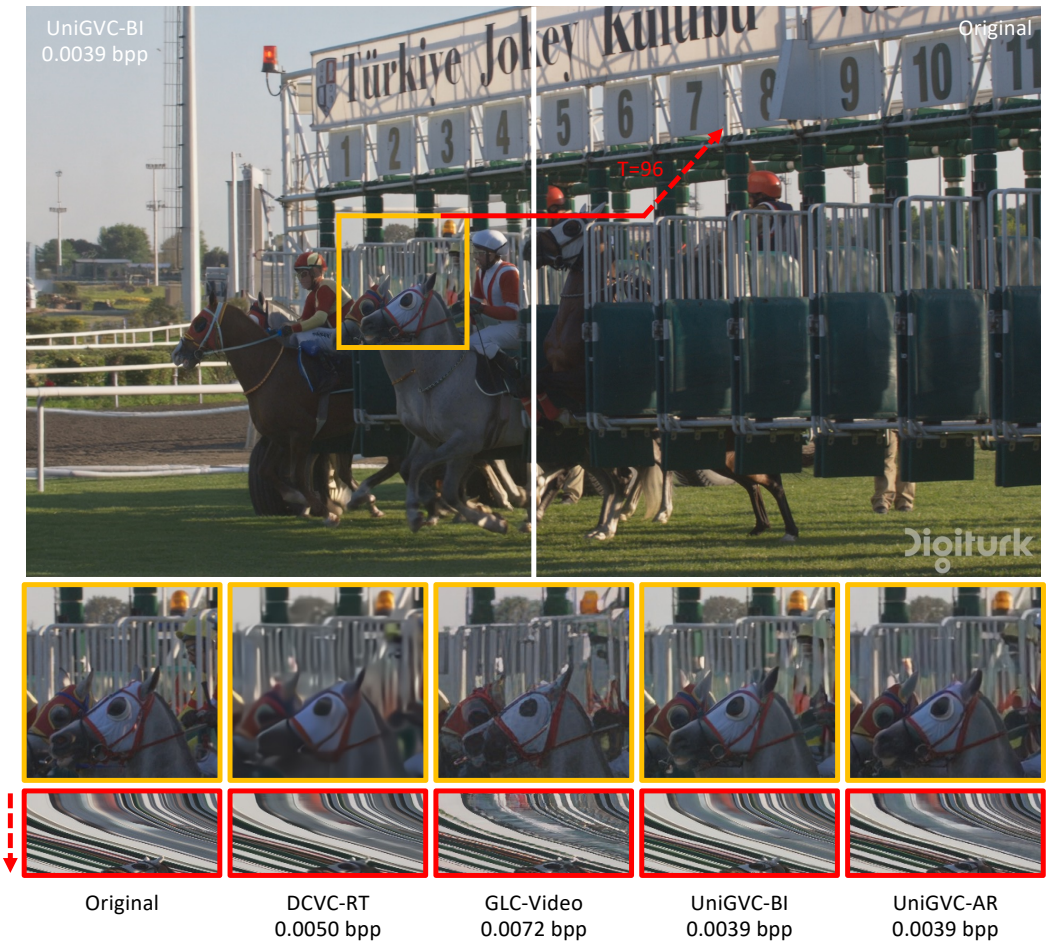

Figure 13: **Visual example from UVG.** *Best viewed on screen for details.*

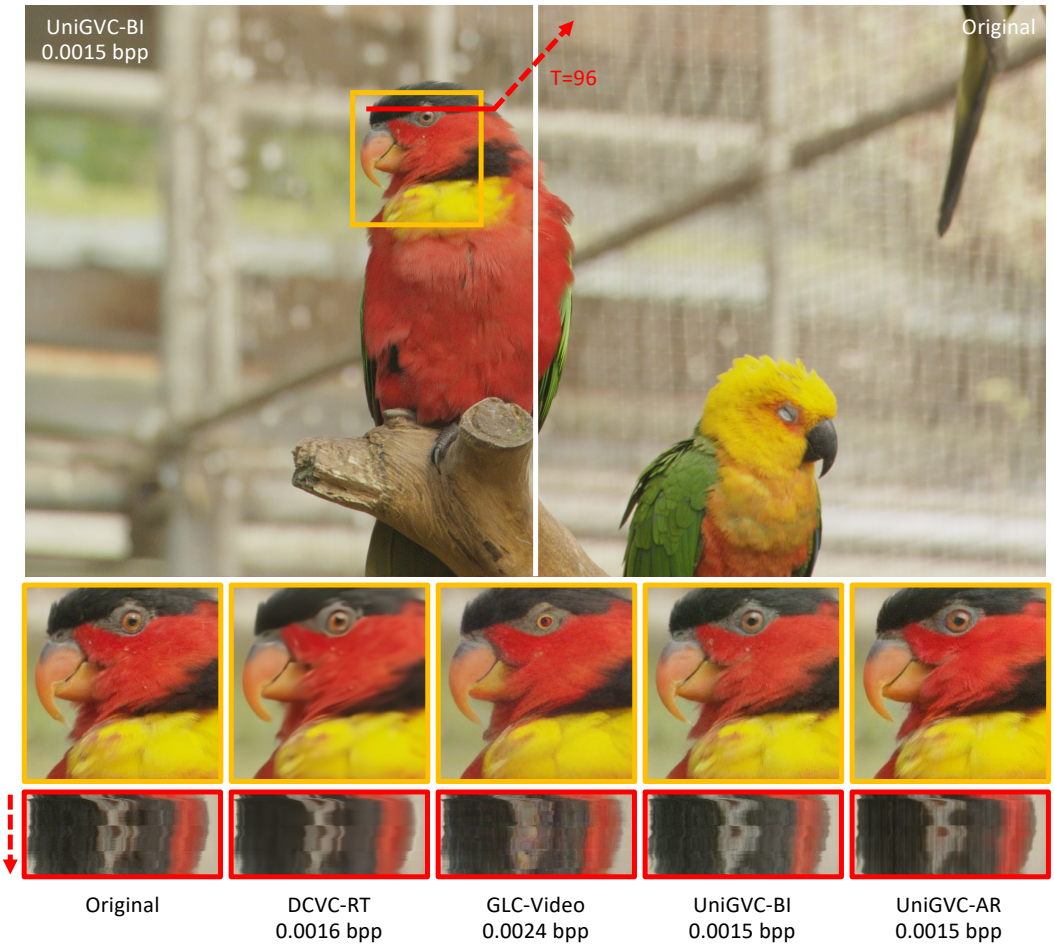

Figure 14: **Visual example from MCL-JCV.** *Best viewed on screen for details.*

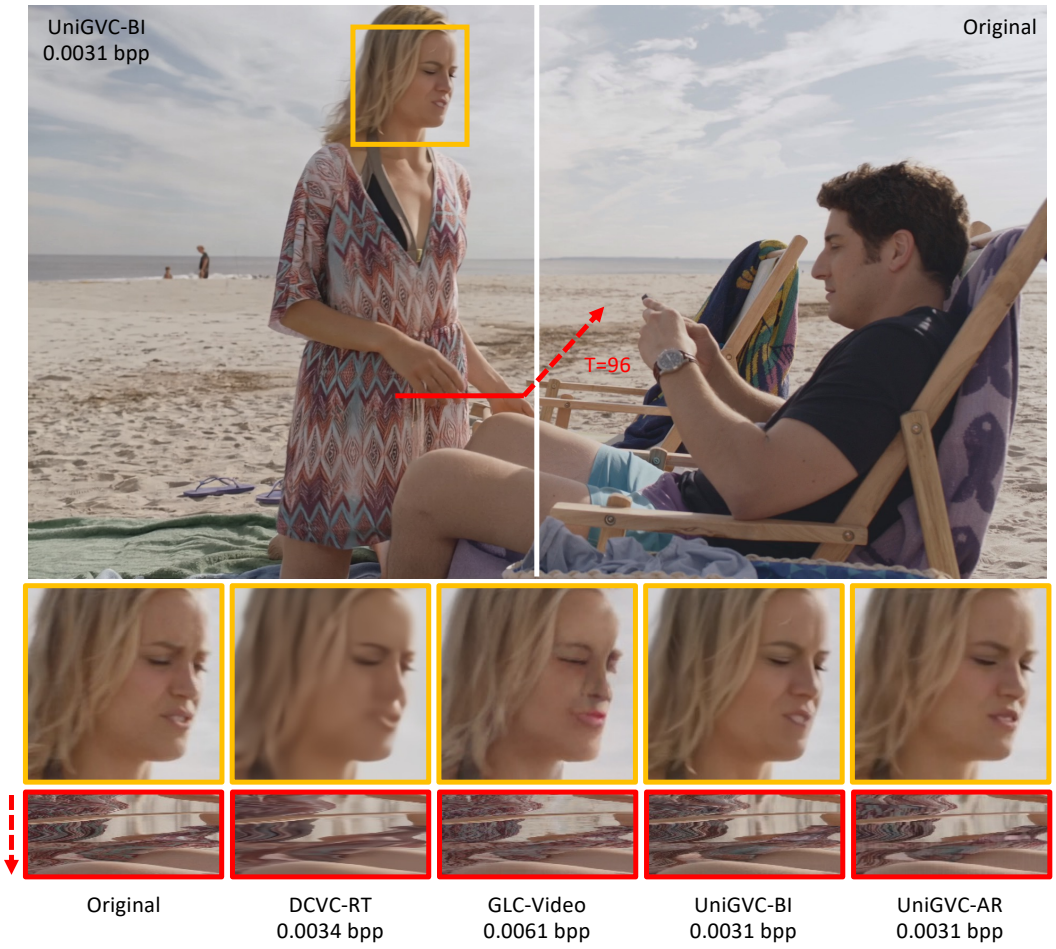

Figure 15: **Visual example from MCL-JCV.** *Best viewed on screen for details.*