GENERATIVE VIDEO COMPRESSION FOR SPATIAL DETAIL AND TEMPORAL CONSISTENCY

Anonymous authors
Paper under double-blind review

ABSTRACT

Generative video compression relies on visual priors to restore detail at low bitrate, while maintaining temporal consistency across frames remains challenging. Frame-wise synthesis with image generative priors can produce inconsistent textures even with temporal conditioning, resulting in flickering artifacts. In this paper, we present GVC-ST, a generative video compression framework for rich spatial detail and stable temporal consistency. GVC-ST divides the video sequence into a reference chain and non-reference frames with distinct responsibilities. The reference chain propagates recovered detail and maintains temporal consistency across the sequence, where each frame is reconstructed by one-step diffusion with forward attention to the preceding reconstructed reference frame. For non-reference frames, each frame is reconstructed from its neighboring references and can optionally be replaced by frame interpolation, achieving high detail without affecting the reference chain. Experimental results on three benchmarks show that GVC-ST achieves the best FID among all compared generative codecs while delivering substantially less temporal flickering than prior generative approaches, as measured by $\mathrm{E}_{\mathrm{warp}}$ and CLIP-F.

1 INTRODUCTION

Neural video compression (NVC) has made substantial progress in recent years (Lu et al., 2019; Li et al., 2021). Recent NVC methods (Jia et al., 2025; Li et al., 2026a) have surpassed traditional standards such as HEVC (Sullivan et al., 2012) and VVC (Bross et al., 2021) in rate–distortion performance. However, these codecs primarily optimize pixel-wise distortion, typically measured by mean squared error (MSE), which does not fully align with human visual perception (Blau & Michaeli, 2019). Meanwhile, limited bit budgets constrain the transmission of fine texture details, and MSE-oriented optimization under these constraints tends to produce overly smooth reconstructions (Mentzer et al., 2020).

To address these problems, generative video compression (GVC) uses learned visual priors to synthesize realistic details and improve perceptual quality at low bitrates. Representative approaches include GAN-based reconstruction (Yang et al., 2022b; Mentzer et al., 2022), generative latent coding (Qi et al., 2025; Guo et al., 2025), and reconstruction with pre-trained diffusion priors (Ma & Chen, 2025a;b). However, maintaining temporal coherence in reconstructed videos remains challenging for many existing methods. Specifically, image generative priors lack explicit temporal modeling, and frame-wise generation can still produce inconsistent textures across frames (Mao et al., 2026) despite previous-frame conditioning and temporal supervision (Yang et al., 2022b; Mentzer et al., 2022). GNVC-VD (Mao et al., 2026) uses the temporal modeling capabilities of a video generative prior to jointly refine spatial and temporal details across frames, improving temporal consistency while maintaining high visual quality.

The coding reference structure also influences detail generation and temporal consistency. In a sequential reference chain, previously reconstructed information guides subsequent reconstruction to reduce temporal flicker (Mentzer et al., 2022; Guo et al., 2025). However, the current frame still needs to update textures according to its own content, and overly strong conditioning on the reference may restrict these updates and limit improvements in perceptual quality. Thus, balancing detail generation and temporal consistency remains challenging under this reference structure. This motivates us to explore whether a different coding structure can improve per-frame perceptual quality while maintaining temporal consistency.

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.

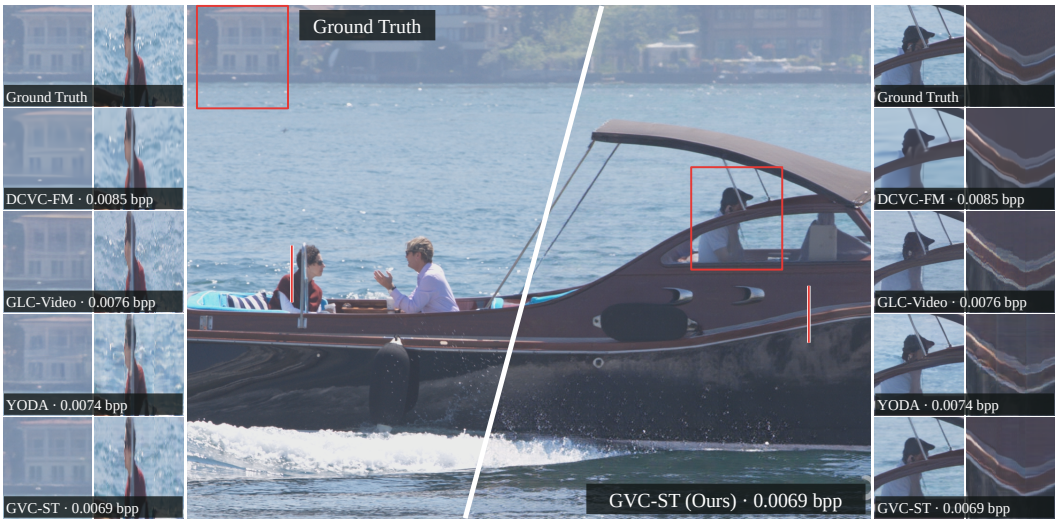


Figure 1: Visual comparisons on YachtRide from UVG. The red boxes highlight local regions for inspecting spatial detail. The space–time slices stack the red-marked pixel columns across consecutive frames, revealing temporal inconsistencies as abrupt changes in color or texture. GVC-ST preserves rich spatial detail while reducing temporal discontinuities at low bitrates.

In this paper, we present GVC-ST, a generative video compression framework for rich spatial detail and stable temporal consistency. GVC-ST assigns distinct roles to reference and non-reference frames. Reference frames propagate recovered details along a persistent reference chain, while non-reference frames use neighboring references to improve spatial detail and local temporal consistency without serving as references for subsequent frames. This separation assigns different reconstruction objectives to the two frame types.

To improve spatiotemporal perceptual quality, we use reconstructed reference latents to guide the recovery of current-frame details. For both reference and non-reference frames, forward reference attention guides one-step diffusion using the latent of the preceding reference. To improve consistency between adjacent frames, we apply bidirectional attention-based latent refinement to non-reference latents.

Since a non-reference frame will not update the reference chain, it can be reconstructed through video frame interpolation (VFI). Reconstructing omitted frames through interpolation has been explored in video compression (Zhang et al., 2013; Wei et al., 2024). However, local interpolation errors may limit MSE-based rate–distortion gains. In our method, high-quality reference frames can improve interpolated detail, and our perceptual objective allows some pixel deviations from the original frame. Interpolation also promotes smooth transitions, supporting temporal consistency. We therefore introduce adaptive non-reference frame synthesis to choose between coding and interpolation for each non-reference frame according to a rate–perception–temporal trade-off.

With role-separated coding and reference-guided generation, GVC-ST achieves the best FID among all compared generative codecs across three benchmarks. It also substantially reduces temporal flickering, as reflected in $\mathrm{E}_{\mathrm{warp}}$ and CLIP-F. Fig. 1 provides a visual comparison of reconstruction detail and temporal continuity at low bitrates.

In summary, our main contributions are as follows:

1. We present GVC-ST, which separates reference propagation from local refinement to preserve spatial detail and temporal consistency.
2. We develop forward reference attention for detail recovery and bidirectional latent refinement for local temporal consistency.
3. We introduce adaptive selection between coding and interpolation for non-reference frames, balancing bitrate, perceptual quality, and temporal consistency.
4. Across three benchmarks, GVC-ST achieves the best FID among compared generative codecs and improves temporal consistency measured by $\mathrm{E}_{\mathrm{warp}}$ and CLIP-F.

2 Related Work

Neural Video Compression. Neural video compression has made substantial progress in rate–distortion performance by learning compact video representations and spatiotemporal dependencies. Representative approaches include motion-compensated residual coding (Lu et al., 2019; Agustsson et al., 2020), 3D spatiotemporal autoencoders (Habibian et al., 2019), and conditional coding (Li et al., 2021). Conditional coding uses feature contexts extracted from decoded references to guide the coding and reconstruction of the current frame, with subsequent work improving coding performance through richer spatiotemporal contexts (Li et al., 2023) and feature modulation (Li et al., 2024). Recent advances also improve B-frame compression through better use of bidirectional references (Liu et al., 2025) and enable real-time coding through implicit temporal modeling (Jia et al., 2025). However, most methods remain primarily optimized for distortion objectives such as MSE, often producing overly smooth reconstructions with missing fine textures at low bitrates. This motivates research into generative priors for improving the perceptual quality of reconstructed videos.

Generative Video Compression. Generative compression uses learned visual priors to reconstruct realistic textures and details at low bitrates. Early work improved the perceptual quality of compressed images through adversarial learning (Mentzer et al., 2020). This approach was later extended to video through spatiotemporal perceptual constraints (Yang et al., 2022b) and mechanisms for synthesizing and propagating details across frames (Mentzer et al., 2022). Meanwhile, other methods exploit the latent representations of generative models for compression. For example, GLC-Video (Qi et al., 2025) compresses video in the latent space of a VQ-VAE, whereas GLVC (Guo et al., 2025) uses continuous latent representations and recurrent memory to retain information from previous frames. Diffusion-based methods offer another approach to perceptual reconstruction. DiffVC (Ma & Chen, 2025a), for instance, combines a pretrained diffusion prior with conditional coding to improve visual quality. However, realistic details in individual frames do not necessarily yield a coherent video. Generated textures must also remain consistent across frames as objects move to avoid distracting flicker. To incorporate temporal information into reconstruction, DiffVC-OSD (Ma & Chen, 2025b) uses temporal context to guide one-step diffusion, while GNVC-VD (Mao et al., 2026) uses a video diffusion prior to improve consistency across frames. Our method addresses this challenge through distinct roles for reference and non-reference frames. Reference frames propagate recovered details, while non-reference frames are refined using neighboring references to maintain both spatial detail and temporal consistency.

Frame Interpolation in Video Coding. Video frame interpolation has been explored to improve rate–distortion performance by reconstructing omitted frames or providing additional references for inter prediction. Early work reduces the encoded frame rate and restores missing frames at the decoder (Zhang et al., 2013). Recent methods also incorporate spatial downsampling and super-resolution (Yang et al., 2022a), and use content-adaptive reconstruction models to improve recovery quality (Wei et al., 2024). The second approach uses interpolated frames as additional reference frames, with representative applications in hierarchical HEVC coding (Guo et al., 2020) and deep reference frame generation for VVC (Jia et al., 2024). Recent work further improves synthesized reference quality through attention mechanisms (Xu et al., 2025). However, under MSE-based optimization, interpolation errors can limit the rate–distortion gains from both frame replacement and reference synthesis. In this work, we investigate when frame interpolation can replace explicit frame coding to improve the efficiency of generative video compression.

3 Methodology

In this work, we aim to preserve rich spatial detail and temporal consistency in low-bitrate generative video compression. Unlike GVC methods that recurrently propagate reference information (Mentzer et al., 2022; Guo et al., 2025), GVC-ST assigns distinct reconstruction roles to reference and non-reference frames. Reference frames propagate recovered details along the reference chain, while non-reference frames undergo local refinement to improve spatial detail and temporal consistency without updating the chain. We first describe the hierarchical coding structure (Section 3.1), followed by reference-guided reconstruction (Section 3.2), adaptive non-reference frame synthesis (Section 3.3), and the training strategy (Section 3.4).

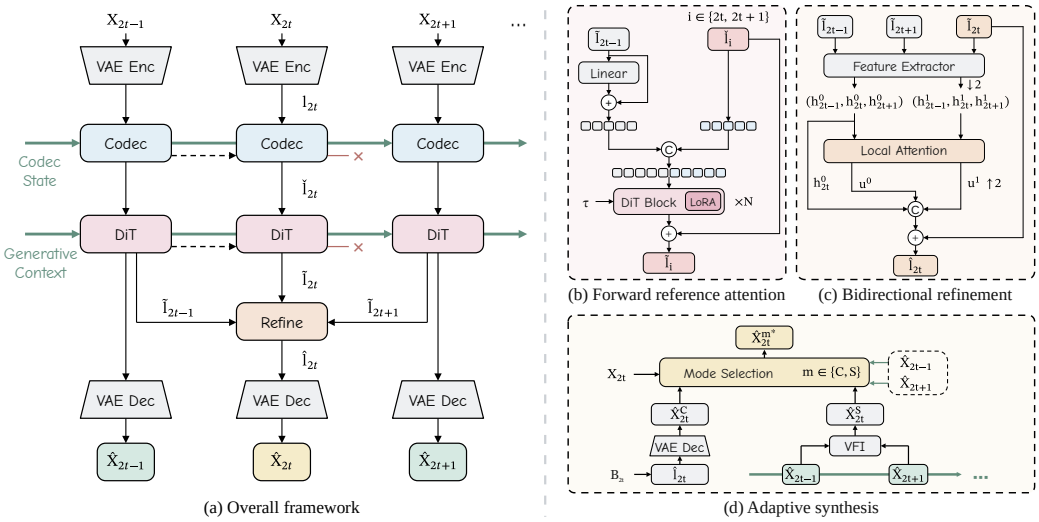


Figure 2: Overview of GVC-ST. (a) Coding and reconstruction at times $2t-1$, $2t$, and $2t+1$. Green and yellow output boxes denote reference and non-reference frames, respectively. Solid green arrows indicate reference propagation, dashed arrows indicate reference use by non-reference frames, and crosses indicate that their states are not propagated. (b) One-step reconstruction with forward reference attention. (c) Latent refinement with bidirectional reference attention. (d) Adaptive selection between coding and interpolation for non-reference frames.

3.1 Hierarchical Coding Structure

As illustrated in Fig. 2(a), GVC-ST maps each input frame $\mathbf{X}_i$ to a latent $\mathbf{l}_i$ using a frozen FLUX.2 VAE encoder (Black Forest Labs, 2026), forming a latent sequence $\mathbf{L}$:

$$\mathbf{L} = \{\mathbf{l}_i\}_{i=0}^{T}, \quad \mathbf{l}_i = \mathcal{E}_{\mathrm{VAE}}(\mathbf{X}_i), \tag{1}$$

where i is the frame index, and T is the number of inter frames following the initial intra frame at $i=0$. Inter frames alternate between reference and non-reference roles, starting with a reference frame at $i=1$. The first inter frame uses the initial intra frame as its reference. We illustrate this reference structure using three frames at consecutive times $2t-1$, $2t$, and $2t+1$. The reference frame at $2t-1$ provides codec state and generative context for the non-reference frame at $2t$ and the next reference frame at $2t+1$.

To reduce inter-latent redundancy, a DCVC-DC-derived latent codec (Li et al., 2023) compresses $\mathbf{l}_i$ using the reconstructed $\check{\mathbf{l}}_{2t-1}$ as its condition:

$$\check{\mathbf{l}}_i, \mathrm{B}_i = \mathrm{Codec}\big(\mathbf{l}_i \mid \check{\mathbf{l}}_{2t-1}\big), \quad i \in \{2t, 2t+1\}, \tag{2}$$

where Codec denotes conditional encoding and decoding, $\check{\mathbf{l}}_i$ is the reconstructed latent, and B_i is the bitstream. For brevity, the codec condition is represented by the reconstructed reference latent. One-step diffusion with a pre-trained FLUX.2 DiT (Black Forest Labs, 2026) then produces the enhanced latent $\tilde{\mathbf{l}}_i$, using the enhanced $\tilde{\mathbf{l}}_{2t-1}$ as its condition:

$$\tilde{\mathbf{l}}_i = \mathrm{DiT}\Big(\check{\mathbf{l}}_i \mid \tilde{\mathbf{l}}_{2t-1}\Big), \quad i \in \{2t, 2t+1\}. \tag{3}$$

Reference latents are decoded directly, while non-reference latents undergo bidirectional refinement before VAE decoding:

$$\hat{\mathbf{X}}_{2t} = \mathcal{D}_{\mathrm{VAE}}(\hat{\mathbf{l}}_{2t}), \quad \hat{\mathbf{X}}_{2t+1} = \mathcal{D}_{\mathrm{VAE}}(\tilde{\mathbf{l}}_{2t+1}), \tag{4}$$

where $\hat{\mathbf{X}}_{2t}$ and $\hat{\mathbf{X}}_{2t+1}$ denote the reconstructed non-reference and reference frames, respectively. The refined latent $\hat{\mathbf{l}}_{2t}$ is obtained using both adjacent reference frames (Section 3.2). The non-reference reconstruction forms the coding candidate for adaptive mode selection in Section 3.3.

This structure propagates recovered details through reference frames while allowing local refinement without changing subsequent reference conditions.

3.2 REFERENCE-GUIDED RECONSTRUCTION

One-step reconstruction with forward reference attention. We use the codec-reconstructed latent $\check{\mathbf{l}}_i$ directly without adding noise. Forward reference attention (Fig. 2(b)) guides texture recovery using $\tilde{\mathbf{l}}_{2t-1}$, which is patchified and passed through a residual linear projection:

$$\mathbf{c}_{2t-1} = \mathrm{Linear}\Big(\mathrm{Patchify}(\tilde{\mathbf{l}}_{2t-1})\Big) + \mathrm{Patchify}(\tilde{\mathbf{l}}_{2t-1}), \tag{5}$$

where $\mathbf{c}_{2t-1}$ denotes the reference tokens. Inspired by ControlNet's zero-initialization strategy (Zhang et al., 2023), the Linear branch is initialized to zero so that these tokens initially preserve the reference patches.

The reference tokens and current-frame tokens obtained by patchifying $\check{\mathbf{l}}_i$ are normalized, concatenated along the sequence dimension, and processed by self-attention in the pre-trained FLUX.2 DiT (Black Forest Labs, 2026). An attention mask allows current tokens to retrieve reference details while blocking attention in the reverse direction. Only the reference projection and LoRA parameters (Hu et al., 2022) are trained, with the base DiT frozen.

Conditioned on $\mathbf{c}_{2t-1}$, the DiT predicts an update at a fixed flow-matching time τ (Lipman et al., 2023). The predicted latent-space update v_θ refines $\check{\mathbf{l}}_i$ through a single residual step:

$$\tilde{\mathbf{l}}_i = \check{\mathbf{l}}_i + (1-\tau) v_\theta(\check{\mathbf{l}}_i, \mathbf{c}_{2t-1}, \tau), \quad i \in \{2t, 2t+1\}, \tag{6}$$

where θ denotes the trainable LoRA parameters.

Latent refinement with bidirectional reference attention. Bidirectional reference attention refines $\tilde{\mathbf{l}}_{2t}$ using $\tilde{\mathbf{l}}_{2t-1}$ and $\tilde{\mathbf{l}}_{2t+1}$. A shared extractor (Fig. 2(c)) produces fine-scale features $\{\mathbf{h}^0_{2t-1}, \mathbf{h}^0_{2t}, \mathbf{h}^0_{2t+1}\}$ and coarse-scale features $\{\mathbf{h}^1_{2t-1}, \mathbf{h}^1_{2t}, \mathbf{h}^1_{2t+1}\}$ from these latents. The fine scale captures local texture details, while the coarse scale provides broader spatial context for cross-frame matching. At each scale s, local cross-attention uses current-frame features $\mathbf{h}^s_{2t}$ as queries and neighboring reference features as keys and values (Vaswani et al., 2017). The retrieved features $\mathbf{r}^s_{2t-1}$ and $\mathbf{r}^s_{2t+1}$ are fused with $\mathbf{h}^s_{2t}$:

$$\mathbf{u}^s = \boldsymbol{\alpha}^s_{2t-1} \odot \mathbf{r}^s_{2t-1} + \boldsymbol{\alpha}^s_{2t} \odot \mathbf{h}^s_{2t} + \boldsymbol{\alpha}^s_{2t+1} \odot \mathbf{r}^s_{2t+1}, \quad s \in \{0, 1\}, \tag{7}$$

where $\mathbf{u}^s$ denotes the fused features, $\boldsymbol{\alpha}$ denotes spatially varying softmax weights shared across channels, and $\odot$ denotes element-wise multiplication. The two reference branches share parameters within each feature level.

The fused features from both scales are combined to predict a bounded residual correction:

$$\hat{\mathbf{l}}_{2t} = \tilde{\mathbf{l}}_{2t} + \tanh\big(\phi\big(\mathrm{Concat}(\mathbf{h}^0_{2t}, \mathbf{u}^0, \mathrm{Upsample}(\mathbf{u}^1))\big)\big), \tag{8}$$

where Upsample resizes $\mathbf{u}^1$ to the resolution of $\mathbf{u}^0$, ϕ denotes the residual prediction head, and $\tanh$ bounds the element-wise correction to $(-1, 1)$.

In summary, reference guidance recovers details and improves local temporal consistency without modifying the reference chain during bidirectional refinement. See Appendix A.2 for implementation details.

3.3 ADAPTIVE NON-REFERENCE FRAME SYNTHESIS

At inference time, the encoder selects either coding or interpolation for each non-reference frame to balance bitrate and reconstruction quality. As shown in Fig. 2(d), the coded candidate is obtained by decoding the refined latent, while RIFE (Huang et al., 2022) generates the interpolated candidate from adjacent reconstructed reference frames:

$$\hat{\mathbf{X}}^{\mathrm{C}}_{2t} = \mathcal{D}_{\mathrm{VAE}}(\hat{\mathbf{l}}_{2t}), \quad \hat{\mathbf{X}}^{\mathrm{S}}_{2t} = \mathrm{VFI}(\hat{\mathbf{X}}_{2t-1}, \hat{\mathbf{X}}_{2t+1}; \tfrac{1}{2}), \tag{9}$$

where C and S denote the coding and interpolation modes, respectively. VFI denotes video frame interpolation, and $1/2$ specifies the temporal midpoint between the two reference frames. Both modes transmit a one-bit mode flag, while only the coding mode transmits the bitstream B_{2t}. Since both modes share the reference frames, their incremental rates are $R_{2t}(\mathrm{C}) = (\ell(\mathrm{B}_{2t}) + 1)/(HW)$ and $R_{2t}(\mathrm{S}) = 1/(HW)$, measured in bpp. Here, H and W denote the frame height and width, and $\ell(\mathrm{B}_{2t})$ is the bitstream length in bits.

The encoder selects the mode as $m^* = \operatorname{argmin}_m J_m$, where $m \in \{\mathrm{C}, \mathrm{S}\}$ denotes the candidate mode. The selection cost jointly accounts for bitrate, perceptual quality, and temporal consistency:

$$\begin{aligned} J_m = R_{2t}(m) + \beta_{\text{perc}} \,\mathrm{LPIPS}(\hat{\mathbf{X}}_{2t}^m, \mathbf{X}_{2t}) \\ + \frac{\beta_{\text{temp}}}{2}\big[\mathrm{E}_{\text{warp}}(\hat{\mathbf{X}}_{2t-1}, \hat{\mathbf{X}}_{2t}^m) + \mathrm{E}_{\text{warp}}(\hat{\mathbf{X}}_{2t}^m, \hat{\mathbf{X}}_{2t+1})\big], \end{aligned} \tag{10}$$

where $\hat{\mathbf{X}}_{2t}^m$ denotes the reconstruction under mode m, and LPIPS (Zhang et al., 2018) measures perceptual distortion against the original frame. E_{warp} measures the mean squared pixel error over non-occluded regions after motion alignment (Lai et al., 2018), using optical flow estimated by RAFT (Teed & Deng, 2020). The weights β_{perc} and β_{temp} control the contributions of perceptual and temporal distortion, respectively.

Overall, modes are selected independently, and the decoder follows the transmitted flag. Neither mode changes subsequent reference conditions.

3.4 Training Strategy

We adopt a three-stage training strategy to preserve structural information, recover perceptual details, and improve local temporal consistency. The VAE and base DiT remain frozen throughout training. Reference and non-reference frames share the codec and one-step reconstruction module.

Stage I: Latent Codec Training. We first train the latent codec with a rate–distortion objective:

$$\mathcal{L}_{\text{codec}} = \frac{1}{T}\sum_{i=1}^{T}\left[R_i + \lambda_{\text{codec}} w_i \,\mathrm{MSE}(\mathcal{D}_{\text{VAE}}(\check{\mathbf{l}}_i), \mathbf{X}_i)\right], \tag{11}$$

where T is the number of inter frames in the training sequence, $\mathbf{X}_i$ is the original frame, R_i is the estimated coding rate in bpp, and w_i is the frame weight. The coefficient λ_{codec} balances distortion against bitrate. Inspired by the hierarchical quality weighting in DCVC-DC (Li et al., 2023), we cycle w_i through $[1.2, 0.5, 0.9, 0.5]$ to encourage allocating more bits to reference frames.

Stage II: One-Step Reconstruction Training. We freeze the codec and train the reference projection and LoRA with MSE for pixel distortion and LPIPS (Zhang et al., 2018) plus DISTS (Ding et al., 2022) for perceptual distortion:

$$\begin{aligned} \mathcal{L}_{\text{base}} = \lambda_{\text{mse}} \,\mathrm{MSE}(\mathcal{D}_{\text{VAE}}(\tilde{\mathbf{l}}_i), \mathbf{X}_i) \\ + \mathrm{LPIPS}(\mathcal{D}_{\text{VAE}}(\tilde{\mathbf{l}}_i), \mathbf{X}_i) + \mathrm{DISTS}(\mathcal{D}_{\text{VAE}}(\tilde{\mathbf{l}}_i), \mathbf{X}_i), \end{aligned} \tag{12}$$

where λ_{mse} weights pixel distortion. We add the generator adversarial loss $\mathcal{L}_{\text{adv}}$ using StableCodec's loss and discriminator (Zhang et al., 2025):

$$\mathcal{L}_{\text{gen}} = \mathcal{L}_{\text{base}} + \lambda_{\text{adv}}\mathcal{L}_{\text{adv}}(\mathcal{D}_{\text{VAE}}(\tilde{\mathbf{l}}_i), \mathbf{X}_i), \tag{13}$$

where λ_{adv} weights adversarial supervision (Goodfellow et al., 2014), introduced after pretraining with $\mathcal{L}_{\text{base}}$.

Stage III: Bidirectional Refinement Training. With the codec and one-step reconstruction module frozen, we train only the bidirectional refiner. For $\hat{\mathbf{X}}_{2t}^{\mathrm{C}} = \mathcal{D}_{\text{VAE}}(\hat{\mathbf{l}}_{2t})$ (Section 3.3), we combine the same base loss with temporal supervision:

$$\begin{aligned} \mathcal{L}_{\text{refine}} = \mathcal{L}_{\text{base}}(\hat{\mathbf{X}}_{2t}^{\mathrm{C}}, \mathbf{X}_{2t}) \\ + \lambda_{\text{temp}}\mathcal{L}_{\text{temp}}\Big(\hat{\mathbf{X}}_{2t}^{\mathrm{C}}, \mathcal{D}_{\text{VAE}}(\tilde{\mathbf{l}}_{2t}), \hat{\mathbf{X}}_{2t-1}, \hat{\mathbf{X}}_{2t+1}\Big), \end{aligned} \tag{14}$$

where λ_{temp} controls the contribution of temporal supervision. The temporal loss compares the motion-aligned errors before and after refinement using E_{warp} (Lai et al., 2018):

$$\begin{aligned} \mathcal{L}_{\text{temp}} &= \tfrac{1}{2}\big[\rho_{2t-1} + \rho_{2t+1} + \max\{\rho_{2t-1}, \rho_{2t+1}\}\big], \\ \rho_i &= \frac{\mathrm{E}_{\text{warp}}(\hat{\mathbf{X}}_{2t}^{\mathrm{C}}, \hat{\mathbf{X}}_i)}{\mathrm{E}_{\text{warp}}(\mathcal{D}_{\text{VAE}}(\tilde{\mathbf{l}}_{2t}), \hat{\mathbf{X}}_i) + \varepsilon}, \quad i \in \{2t-1, 2t+1\}, \end{aligned} \tag{15}$$

where ε ensures numerical stability. References and denominators remain fixed. The loss encourages consistency with both references and emphasizes the larger error ratio.

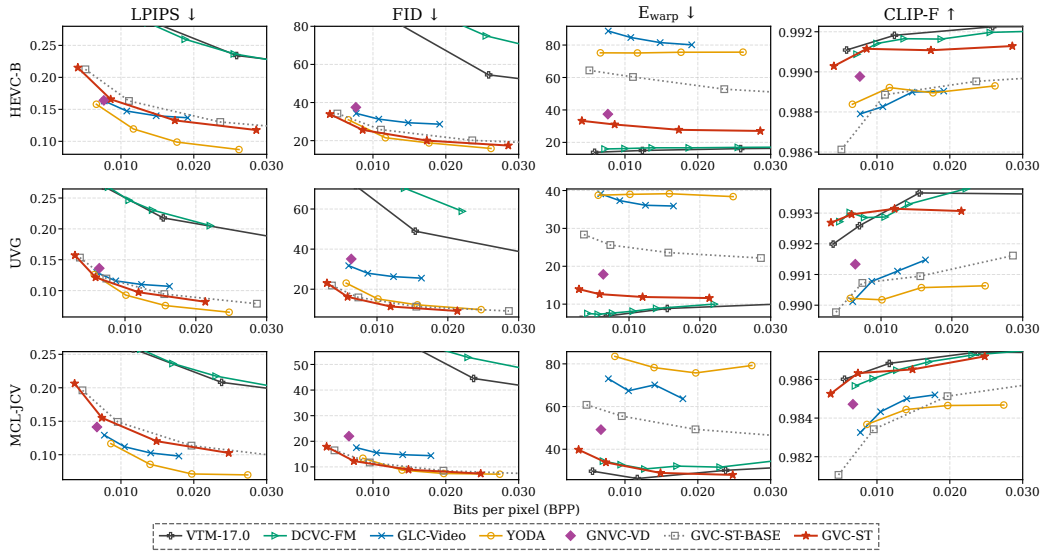


Figure 3: Rate–distortion curves in terms of LPIPS, FID, E_{warp}, and CLIP-F.

Table 1: Results relative to VTM-17.0. LPIPS: BD-Rate (%); other metrics: BD-Metric.

	HEVC-B				UVG				MCL-JCV			
Method	LPIPS↓	FID↓	E_{warp} ↓	CLIP-F↑	LPIPS↓	FID↓	E_{warp} ↓	CLIP-F↑	LPIPS↓	FID↓	E_{warp} ↓	CLIP-F↑
VTM-17.0	0.00	0.00	0.00	0.0000	0.00	0.00	0.00	0.0000	0.00	0.00	0.00	0.0000
DCVC-FM	3.83	20.27	1.13	-0.0003	10.38	19.13	0.48	0.0000	11.66	4.50	2.88	-0.0003
GLC-Video	-93.01	-50.72	68.66	-0.0033	-93.44	-33.42	29.23	-0.0023	-94.06	-45.86	39.10	-0.0024
YODA	-95.82	-57.75	60.15	-0.0028	-95.33	-42.23	30.78	-0.0029	-86.44	-46.27	48.98	-0.0026
GVC-ST-BASE	-89.71	-51.62	41.63	-0.0029	-93.68	-47.73	16.98	-0.0023	-87.60	-48.38	23.09	-0.0028
GVC-ST	-91.63	-57.92	14.16	-0.0007	-94.72	-53.37	4.85	0.0001	-88.89	-51.77	1.50	-0.0003

4 Experiments

4.1 Experimental Setup

Datasets. We use Vimeo-90K (Xue et al., 2019), original Vimeo sequences (Original Vimeo links), and BVI-DVC (Ma et al., 2022) for training. Evaluation uses the first 96 frames of each sequence from HEVC Class B (Bossen, 2013), UVG (Mercat et al., 2020), and MCL-JCV (Wang et al., 2016). Following DCVC-DC (Li et al., 2023), the original YUV420 videos are converted to RGB using BT.709 (ITU-R, 2015), with all quality metrics computed in RGB space.

Implementation Details. We build the latent codec upon DCVC-DC (Li et al., 2023) and use the pre-trained VAE and DiT from FLUX.2 [klein] Base 4B (Black Forest Labs, 2026). Training follows the three stages in Section 3.4. In Stage I, the latent codec uses rate–distortion weights $\lambda_{\text{codec}} \in \{2, 8, 32, 96\}$. In Stage II, we use rank-96 LoRA and set $\lambda_{\text{mse}} = 10$ for reconstruction pretraining and $\lambda_{\text{mse}} = 1$ for adversarial fine-tuning. During adversarial fine-tuning, λ_{adv} gradually increases to 0.1 after an initial warm-up. In Stage III, the bidirectional refiner uses $\lambda_{\text{mse}} = 10$ and $\lambda_{\text{temp}} = 0.2$. One-step reconstruction uses a fixed time condition $\tau = 0.05$ and a fixed empty-prompt embedding. During inference, adaptive synthesis uses RIFE (Huang et al., 2022) for interpolation, with selection weights $\beta_{\text{perc}} = 0.68$ and $\beta_{\text{temp}} = 0.02$.

Compared Methods. We compare GVC-ST with the traditional codec VTM-17.0 and the MSE-optimized neural codec DCVC-FM (Li et al., 2024). Generative baselines include GLC-Video (Qi et al., 2025) and YODA (Li et al., 2026b). The learned baselines use publicly released pre-trained models. Following DCVC-DC (Li et al., 2023), an intra period of 32 is used for all these codecs, including GVC-ST. VTM uses its official low-delay configuration (JVET, 2022) with 10-bit YUV444 coding. Its decoded frames are converted to RGB for evaluation. We additionally reimplement GNVC-VD (Mao et al., 2026) following its reported architecture and training procedure. Due to



Figure 4: Spatial detail comparison on BasketballDrive.

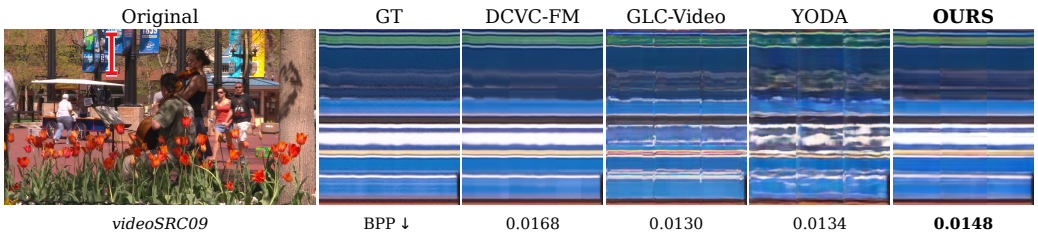


Figure 5: Temporal consistency comparison on videoSRC09. The space–time slices stack the marked pixel column across 96 consecutive frames, with the horizontal axis representing time.

computational constraints, we train GNVC-VD at a single rate setting and report the corresponding operating point on each test set. To assess the contributions of our proposed modules, we also introduce GVC-ST-BASE. This baseline is derived from GVC-ST, but disables forward reference attention, bidirectional latent refinement, and adaptive synthesis.

Evaluation Metrics. We evaluate perceptual quality using LPIPS-Alex (Zhang et al., 2018) and FID (Heusel et al., 2017). Temporal consistency is assessed using E_{warp} (Lai et al., 2018) and CLIP-F based on CLIP features (Radford et al., 2021). E_{warp} measures motion-compensated pixel consistency, while CLIP-F evaluates semantic continuity across frames. Lower values indicate better quality for all metrics except CLIP-F. Rate is measured in bits per pixel (bpp). For the main comparison, we use VTM-17.0 as the anchor and summarize multi-rate results using Bjøntegaard curve integration (Bjøntegaard, 2001). For LPIPS, we report BD-Rate, the average percentage bitrate change at equal perceptual distortion. Negative values indicate bitrate savings. For FID, E_{warp}, and CLIP-F, we report BD-Metric, the average metric difference at equal bitrate over the common log-rate interval. Negative values indicate improvements in FID and E_{warp}, while positive values indicate improvements in CLIP-F.

4.2 Comparison with Representative Codecs

Quantitative Comparison. Figure 3 and Table 1 summarize the rate–distortion results. Relative to VTM-17.0, GVC-ST achieves LPIPS BD-Rate savings of 91.6%, 94.7%, and 88.9% on HEVC-B, UVG, and MCL-JCV, respectively. It also achieves the best FID BD-Metric among the compared generative codecs. Compared with GNVC-VD, GVC-ST shows greater perceptual realism at comparable bitrates. Beyond spatial perceptual quality, GVC-ST provides stronger temporal consistency than GLC-Video and YODA across all three datasets. This advantage is also observed over GNVC-VD at comparable bitrates. Relative to GVC-ST-BASE, the full model improves all four metrics on each dataset, supporting the combined effectiveness of the proposed modules.

Qualitative Comparison. Figure 4 compares representative methods at low bitrates. In the BasketballDrive example, GVC-ST reconstructs finer details at only 0.0043 bpp, the lowest bitrate among the displayed methods. DCVC-FM produces visibly smoother textures, while GLC-Video and YODA introduce detail distortions. Figure 5 further illustrates temporal consistency through space–time slices. GLC-Video and YODA exhibit abrupt texture changes and irregular edges over time. In comparison, GVC-ST produces fewer texture discontinuities and more continuous edges. These examples illustrate its ability to recover spatial details while maintaining temporal consistency at low bitrates.

Table 2: Ablation study on MCL-JCV. All quality metrics are BD-Metrics relative to A.

Variant	Configuration			BD-Metrics			
	Forward Attention	Bidirectional Refinement	Synthesis Mode	LPIPS↓	E_{warp} ↓	$\text{E}_{\text{warp}}^{\text{Ref}}$ ↓	$\text{E}_{\text{warp}}^{\text{NonRef}}$ ↓
A(GVC-ST-BASE)			Coding	0.0000	0.00	0.00	0.00
B	✓		Coding	-0.0009	-0.43	-2.77	5.19
C	✓	✓	Coding	-0.0010	-3.96	-2.77	-11.57
D	✓		Interpolation	-0.0053	-19.61	-5.59	-31.55
E(GVC-ST)	✓	✓	Adaptive	-0.0073	-22.89	-5.38	-34.03



Figure 6: Adaptive synthesis examples. Coding is selected in the first example and interpolation in the second. Zoom in to inspect local details.

4.3 Ablation Studies

We evaluate the proposed modules on MCL-JCV, with results summarized in Table 2. LPIPS measures perceptual quality, while E_{warp} measures temporal consistency between adjacent frames. $\text{E}_{\text{warp}}^{\text{Ref}}$ measures consistency between consecutive reference frames, while $\text{E}_{\text{warp}}^{\text{NonRef}}$ measures consistency between consecutive non-reference frames. All four metrics are reported as BD-Metrics relative to GVC-ST-BASE (A), with negative values indicating improvements.

Reference-Guided Reconstruction. Forward reference attention (B) improves perceptual quality and reference-frame consistency over A, but increases temporal error between non-reference frames. To address this, bidirectional refinement (C) uses both neighboring references to refine non-reference frames. Compared with B, it reduces non-reference-frame and whole-sequence warping errors while maintaining perceptual quality. This refinement requires no additional bitrate and leaves reference-frame reconstruction unchanged.

Adaptive Non-reference Frame Synthesis. Replacing non-reference-frame coding with fixed interpolation (D) improves perceptual rate–distortion performance over B. However, the first example in Figure 6 reveals a failure case: the character undergoes a substantial pose change between reference frames, and interpolation produces distorted limbs and blurred boundaries. GVC-ST (E) selects coding to preserve the current frame's structure in this case. In the second example, interpolation achieves comparable perceptual quality with lower warping error while avoiding the coded frame payload. GVC-ST (E) selects interpolation in this case. By combining refined coded candidates with adaptive selection, GVC-ST (E) achieves lower LPIPS and whole-sequence warping error than D.

5 Conclusion

We presented GVC-ST for generative video compression, with distinct roles for reference and non-reference frames. Forward reference attention propagates recovered details along the reference chain, while bidirectional refinement improves non-reference frames without changing subsequent reference conditions. Adaptive selection between coding and interpolation further improves perceptual rate–distortion performance. Results on three benchmarks demonstrate improvements in perceptual quality and temporal consistency at low bitrates. These findings highlight the value of coordinating reference propagation and non-reference-frame synthesis for temporally consistent generative video compression.

Limitations. GVC-ST remains computationally demanding. As detailed in Appendix A.4, the DiT dominates the computational cost. Generating both coding and interpolation candidates introduces additional overhead. Future work will explore more efficient generative backbones and mode selection without generating both candidates.

AI USE STATEMENT.

We used generative AI tools to assist with manuscript editing, including wording, grammar, clarity, and terminology consistency. These tools also assisted with formatting tables of existing experimental results and editing scientific figures. The authors take full responsibility for the accuracy of the reported data and the final content of the paper, including all AI-assisted text and materials.

REFERENCES

Eirikur Agustsson, David Minnen, Nick Johnston, Johannes Ballé, Sung Jin Hwang, and George Toderici. Scale-space flow for end-to-end optimized video compression. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8503–8512, 2020.

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

Gisle Bjøntegaard. Calculation of average PSNR differences between RD-curves. Technical Report VCEG-M33, ITU-T Video Coding Experts Group, 2001.

Black Forest Labs. FLUX.2 [klein]: Towards interactive visual intelligence. `https://bfl.ai/blog/flux2-klein-towards-interactive-visual-intelligence`, 2026.

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

Frank Bossen. Common test conditions and software reference configurations. Technical Report JCTVC-L1100, Joint Collaborative Team on Video Coding, 2013.

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.

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

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.

Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in Neural Information Processing Systems*, volume 27, 2014.

Yu Guo, Zizheng Liu, Zhenzhong Chen, and Shan Liu. Deep inter coding with interpolated reference frame for hierarchical coding structure. In *IEEE International Conference on Visual Communications and Image Processing*, pp. 302–305, 2020.

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

Amirhossein Habibian, Ties van Rozendaal, Jakub M. Tomczak, and Taco S. Cohen. Video compression with rate-distortion autoencoders. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 7033–7042, 2019.

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*, volume 30, 2017.

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*, 2022.

Zhewei Huang, Tianyuan Zhang, Wen Heng, Boxin Shi, and Shuchang Zhou. Real-time intermediate flow estimation for video frame interpolation. In *European Conference on Computer Vision*, pp. 624–642. Springer, 2022.

ITU-R. Parameter values for the HDTV standards for production and international programme exchange. Technical Report Recommendation ITU-R BT.709-6, International Telecommunication Union, 2015. URL https://www.itu.int/rec/R-REC-BT.709-6-201506-I.

Jianghao Jia, Yuantong Zhang, Han Zhu, Zhenzhong Chen, Zizheng Liu, Xiaozhong Xu, and Shan Liu. Deep reference frame generation method for VVC inter prediction enhancement. *IEEE Transactions on Circuits and Systems for Video Technology*, 34(5):3111–3124, 2024.

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*, pp. 12543–12552, 2025.

JVET. VVC test model (VTM), version 17.0. https://vcgit.hhi.fraunhofer.de/jvet/VVCSoftware_VTM/-/tags/VTM-17.0, 2022. Release VTM-17.0, commit e87b8ffa.

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. In *Advances in Neural Information Processing Systems*, volume 34, pp. 18114–18125, 2021.

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

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

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*, 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. Early access.

Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. In *International Conference on Learning Representations*, 2023.

Yuxi Liu, Dengchao Jin, Shuai Huo, Jiawen Gu, Chao Zhou, Huihui Bai, Ming Lu, and Zhan Ma. Neural B-frame video compression with bi-directional reference harmonization. In *Advances in Neural Information Processing Systems*, volume 38, 2025.

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*, pp. 11006–11015, 2019.

Di Ma, Fan Zhang, and David R. Bull. BVI-DVC: A training database for deep video compression. *IEEE Transactions on Multimedia*, 24:3847–3858, 2022.

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):345:1–345:22, 2025a.

Wenzhuo Ma and Zhenzhong Chen. DiffVC-OSD: One-step diffusion-based perceptual neural video compression framework. In *IEEE International Conference on Visual Communications and Image Processing*, pp. 1–5, 2025b.

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*, pp. 43239–43248, 2026.

Fabian Mentzer, George D. Toderici, Michael Tschannen, and Eirikur Agustsson. High-fidelity generative image compression. In *Advances in Neural Information Processing Systems*, volume 33, pp. 11913–11924, 2020.

Fabian Mentzer, Eirikur Agustsson, Johannes Ballé, David Minnen, Nick Johnston, and George Toderici. Neural video compression using GANs for detail synthesis and propagation. In *European Conference on Computer Vision*, pp. 562–578. Springer, 2022.

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.

Original Vimeo links. `https://github.com/anchen1011/toflow/blob/master/data/original_vimeo_links.txt`, n.d.

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.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139, pp. 8748–8763. PMLR, 2021.

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.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30, 2017.

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 *IEEE International Conference on Image Processing*, pp. 1509–1513, 2016.

Zhou Wang, Eero P. Simoncelli, and Alan 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.

Yiying Wei, Hadi Amirpour, Ahmed Telili, Wassim Hamidouche, Guo Lu, and Christian Timmerer. Video encoding enhancement via content-aware spatial and temporal super-resolution. In *European Signal Processing Conference*, pp. 681–685, 2024.

Linchen Xu, Zhikai Liu, and Fan Liang. Attention based deep reference frame enhancement for VVC inter prediction. In *Asia Pacific Signal and Information Processing Association Annual Summit and Conference*, pp. 2436–2441, 2025.

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, 2019.

Jiayu Yang, Chunhui Yang, Fei Xiong, Feng Wang, and Ronggang Wang. Learned low bitrate video compression with space-time super-resolution. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pp. 1786–1790, 2022a.

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

Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 3836–3847, 2023.

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 Conference on Computer Vision and Pattern Recognition*, pp. 586–595, 2018.

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

Yongbing Zhang, Haoqian Wang, and Debin Zhao. Up-sampling oriented frame rate reduction. *Signal Processing: Image Communication*, 28(3):254–266, 2013.

A APPENDIX

A.1 TEST SETTINGS

Traditional Codecs. VTM-17.0 uses the official low-delay configuration, *encoder_lowdelay_vtm.cfg* (JVET, 2022), with 10-bit YUV444 input and an intra period of 32. The QP values are 17, 22, 27, 32, 37, and 42. The encoder is invoked with the following options:

- -c *encoder_lowdelay_vtm.cfg*
  --InputFile={*input video*}
  --InputBitDepth=10
  --OutputBitDepth=10
  --OutputBitDepthC=10
  --InputChromaFormat=444
  --FrameRate={*frame rate*}
  --DecodingRefreshType=2
  --FramesToBeEncoded=96
  --SourceWidth={*width*}
  --SourceHeight={*height*}
  --IntraPeriod=32
  --QP={*qp*}
  --Level=6.2
  --BitstreamFile={*bitstream file*}
  --ReconFile={*reconstructed video*}

Decoded frames are converted to RGB using BT.709 (ITU-R, 2015) for quality evaluation. Bitrate is computed from the complete output bitstream.

Neural Codecs. The configurations used for neural codecs are summarized below:

- **DCVC-FM** (Li et al., 2024). We use the official pre-trained weights with quantization indices 22, 27, 32, 37, 42, and 49. The intra period and feature reset interval are both set to 32, with a rate-control GOP size of 8.
- **GLC-Video** (Qi et al., 2025). We use the public implementation and pre-trained models with quantization indices 0, 21, 42, and 63. The intra period and feature reset interval are both set to 32.
- **YODA** (Li et al., 2026b). We use the official pre-trained weights for four rate points. The intra period is set to 32.
- **GNVC-VD** (Mao et al., 2026). We retrain the model following the original architecture, training procedure, and coding configuration, and evaluate one operating point per dataset.
- **GVC-ST.** Both GVC-ST and GVC-ST-BASE use an intra period of 32 and rate indices 0, 21, 42, and 63.

A.2 IMPLEMENTATION DETAILS

Given a frame $\mathbf{X}_i \in \mathbb{R}^{H\times W\times 3}$, the frozen FLUX.2 [klein] Base 4B VAE encoder (Black Forest Labs, 2026) maps it to a latent $\mathbf{l}_i \in \mathbb{R}^{(H/8)\times(W/8)\times 32}$.

Intra-frame coding and reconstruction. The intra-frame codec adapts the image codec of DCVC-DC (Li et al., 2023) to VAE latents. We modify its input and output layers to accommodate 32-channel latents. Since the VAE encoder has already reduced the frame resolution, we also reduce the spatial downsampling factor from 16 to 2. The hyperprior and spatial context model are retained for entropy modeling. After codec decoding, a separately adapted FLUX.2 DiT enhances the reconstructed latent $\check{\mathbf{l}}_0$ in one step without reference tokens, producing $\tilde{\mathbf{l}}_0$. This latent is then decoded by the VAE decoder. Finally, the codec state and generative context are passed to subsequent inter frames.

Inter-frame coding. Figure 7 shows the DCVC-DC-derived inter-frame codec (Li et al., 2023). As in the intra-frame codec, we adapt the input and output layers to 32-channel latents and reduce the spatial downsampling factor from 16 to 2. The reference state at time $2t-1$ provides temporal contexts $\mathbf{C}^0_{2t-1}$ and $\mathbf{C}^1_{2t-1}$ for conditional coding. For $i \in \{2t, 2t+1\}$, g_a maps $\mathbf{l}_i$ to coding features

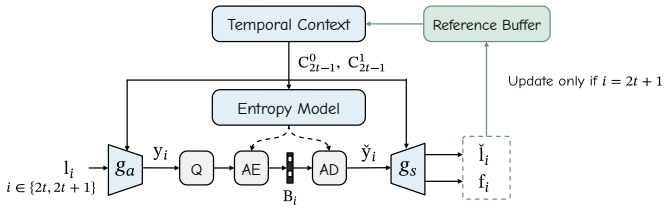
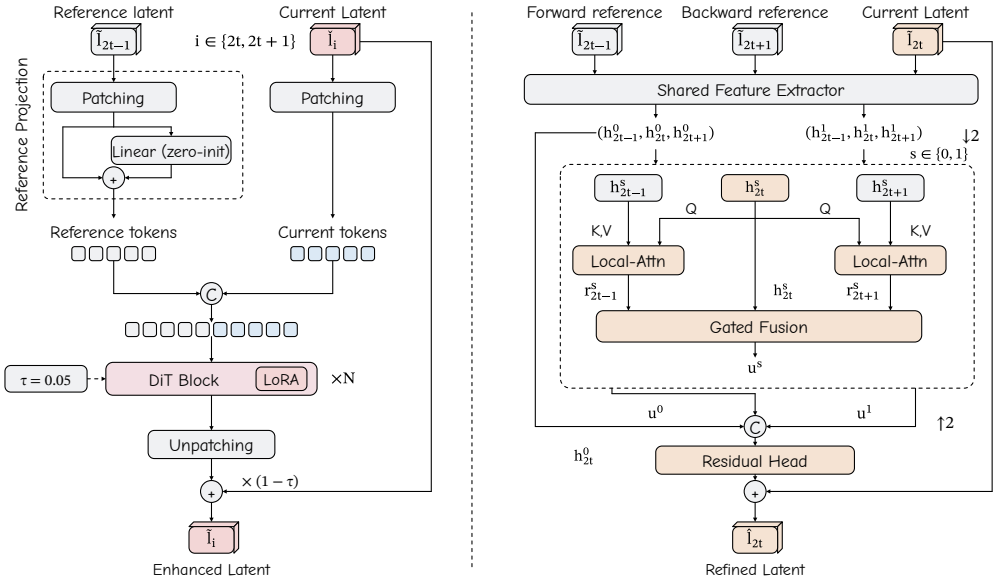


Figure 7: Inter-frame latent codec. Q: quantization; AE/AD: arithmetic encoding/decoding; g_a/g_s: analysis/synthesis transforms.



(a) One-Step Reconstruction with Forward Attention

(b) Latent Refinement with Bidirectional Reference Attention

Figure 8: Reference-guided reconstruction: (a) one-step reconstruction with forward reference attention; (b) latent refinement with bidirectional reference attention.

$\mathbf{y}_i$, which are quantized and entropy-coded into B_i. After entropy decoding, g_s maps the quantized features $\check{\mathbf{y}}_i$ to the reconstructed latent $\check{\mathbf{I}}_i$ and coding features $\mathbf{f}_i$. These outputs update the reference buffer only when $i = 2t + 1$.

One-step reconstruction with forward reference attention. Figure 8(a) enhances the codec-reconstructed latent $\check{\mathbf{I}}_i$ using the preceding reference latent $\tilde{\mathbf{I}}_{2t-1}$. Both latents are partitioned into 2×2 patches. A zero-initialized residual linear projection converts the reference patches into conditioning tokens. The reference and current tokens are then normalized using the pre-trained statistics and concatenated. An attention mask allows current tokens to retrieve reference details while preventing reference tokens from attending to current tokens. The DiT predicts updates for the current tokens. These predictions undergo inverse normalization and unpatchifying to obtain the latent-space update. This update is scaled by $1 - \tau$ and added to $\check{\mathbf{I}}_i$ to produce $\tilde{\mathbf{I}}_i$, with $\tau = 0.05$.

Latent refinement with bidirectional reference attention. Figure 8(b) refines the non-reference latent $\tilde{\mathbf{I}}_{2t}$ using the adjacent enhanced reference latents $\tilde{\mathbf{I}}_{2t-1}$ and $\tilde{\mathbf{I}}_{2t+1}$. A shared extractor produces features at the original and half latent resolutions. At each scale, current features provide queries, while reference features provide keys and values for local attention. The forward and backward branches share parameters, while different scales use independent parameters. Gated fusion combines the retrieved reference features with the current features. The coarse fused feature is then upsampled and concatenated with the fine fused feature and the current-frame fine-scale feature. A residual head predicts a correction, which is bounded by $\tanh$. The correction is added to $\tilde{\mathbf{I}}_{2t}$ to obtain $\hat{\mathbf{I}}_{2t}$. Inspired

Table 3: Additional results relative to VTM-17.0. DISTS: BD-Rate (%); other metrics: BD-Metric.

	HEVC-B					UVG					MCL-JCV				
Method	DISTS↓	KID↓	PSNR↑	MS-SSIM↑	FloLPIPS↓	DISTS↓	KID↓	PSNR↑	MS-SSIM↑	FloLPIPS↓	DISTS↓	KID↓	PSNR↑	MS-SSIM↑	FloLPIPS↓
VTM-17.0	0.00	0.0000	0.00	0.0000	0.0000	0.00	0.0000	0.00	0.0000	0.0000	0.00	0.0000	0.00	0.0000	0.0000
DCVC-FM	37.80	0.0109	0.38	0.0037	0.0028	39.85	0.0049	0.40	0.0017	0.0082	34.34	-0.0004	0.20	0.0003	0.0081
GLC-Video	-90.04	-0.0327	-5.99	-0.0902	-0.1560	-93.18	-0.0186	-6.15	-0.0677	-0.1453	-92.16	-0.0275	-5.54	-0.0557	-0.1476
YODA	-98.08	-0.0329	-6.50	-0.1267	-0.1816	N/A	-0.0190	-6.81	-0.0977	-0.1604	N/A	-0.0251	-6.40	-0.0802	-0.1550
GVC-ST-BASE	-86.24	-0.0303	-4.77	-0.0631	-0.1352	-94.59	-0.0212	-4.04	-0.0448	-0.1471	-87.32	-0.0270	-4.16	-0.0413	-0.1150
GVC-ST	-88.34	-0.0337	-4.23	-0.0494	-0.1518	-95.40	-0.0234	-3.74	-0.0383	-0.1599	-89.25	-0.0290	-4.26	-0.0410	-0.1238

BD integration uses cubic fits over each target–anchor overlap without extrapolation. N/A denotes fewer than four unique points or overlap below 50% of the target span (quality for BD-Rate, log-rate for BD-Metric).

Table 4: Module-wise parameters and computation for inter-frame processing in GVC-ST.

Module	Params (M)	KMACs/pixel
VAE encoder	34.43	2609.85
VAE decoder	49.62	8025.97
Latent codec	18.05	582.14
DiT	3958.12	41226.17
Forward reference projection	0.02	0.06
Bidirectional latent refinement	0.58	6.41
RIFE	5.66	11.47
Total	**4066.48**	**52462.07**

by ControlNet's zero-initialization strategy (Zhang et al., 2023), we initialize the output projection to zero so that the refiner initially preserves its input.

A.3 Additional Rate–Distortion Results

For a more comprehensive comparison, Figure 9 and Table 3 present additional rate–distortion results for GVC-ST and the baseline methods using DISTS (Ding et al., 2022), KID (Bińkowski et al., 2018), PSNR, MS-SSIM (Wang et al., 2003), and FloLPIPS (Danier et al., 2022). Compared with generative codecs GLC-Video (Qi et al., 2025) and YODA (Li et al., 2026b), GVC-ST achieves better KID BD-Metrics across all three datasets, alongside improvements in the distortion-oriented metrics PSNR and MS-SSIM. Relative to GVC-ST-BASE, it also improves DISTS, KID, and FloLPIPS on all three datasets. These results complement the perceptual quality and temporal consistency comparisons in the main paper.

A.4 Complexity

We analyze the computational complexity of GVC-ST in terms of model size and computation for inter-frame processing. As summarized in Table 4, the inter-frame pipeline contains 4066.48M parameters, including 84.05M in the VAE, 18.05M in the latent codec, and 3958.12M in the DiT. The VAE and DiT account for most of the computational cost. Future work will focus on improving their efficiency.

A.5 Additional Visual Comparisons

Figure 10 compares local reconstructions on five sequences using the closest available baseline bitrates to GVC-ST. GVC-ST preserves fine details at lower bitrates than the other displayed methods in all five examples. These results complement the spatial detail comparisons in the main paper.

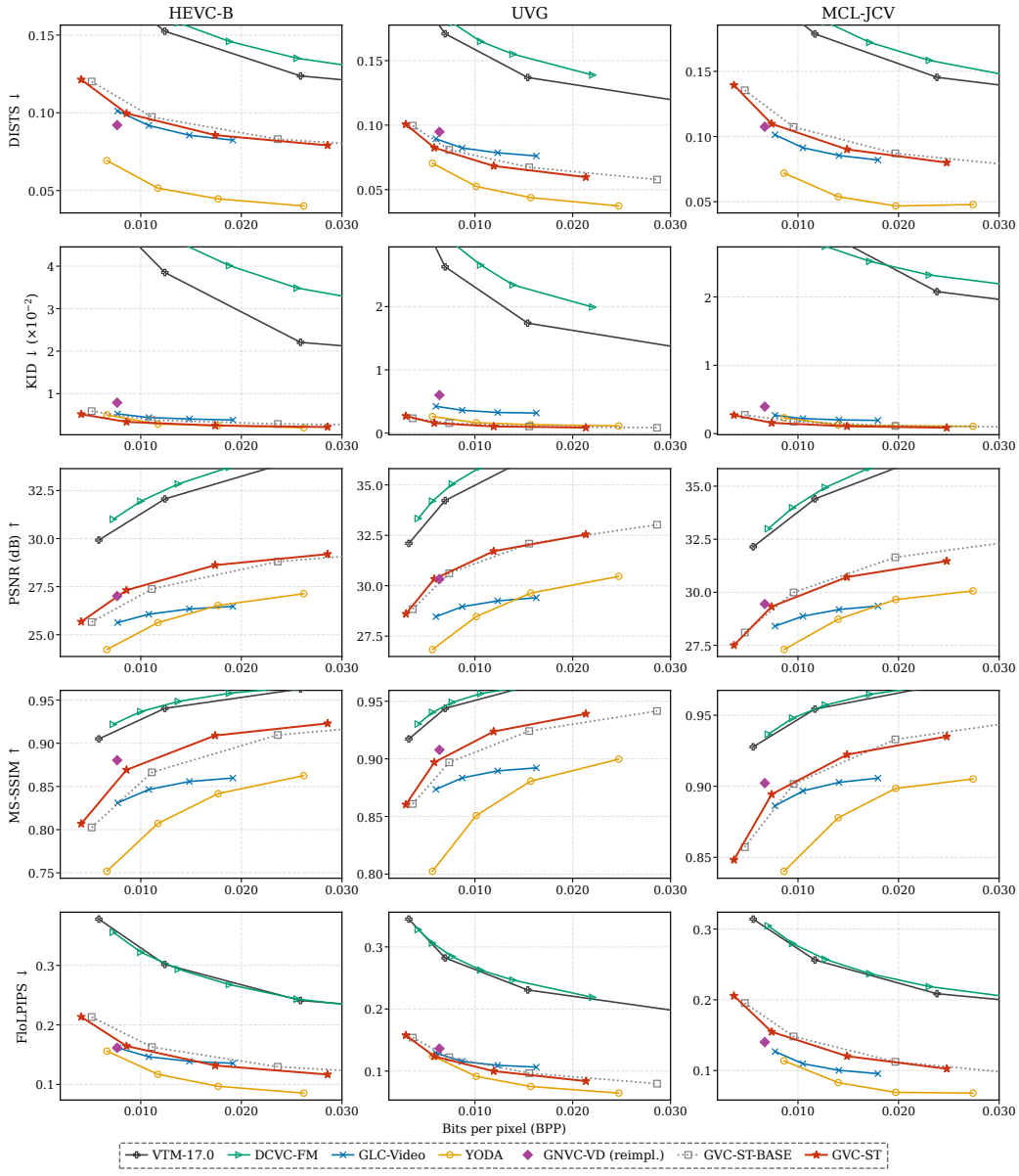


Figure 9: Rate–distortion curves in terms of DISTS, KID, PSNR, MS-SSIM, and FloLPIPS.

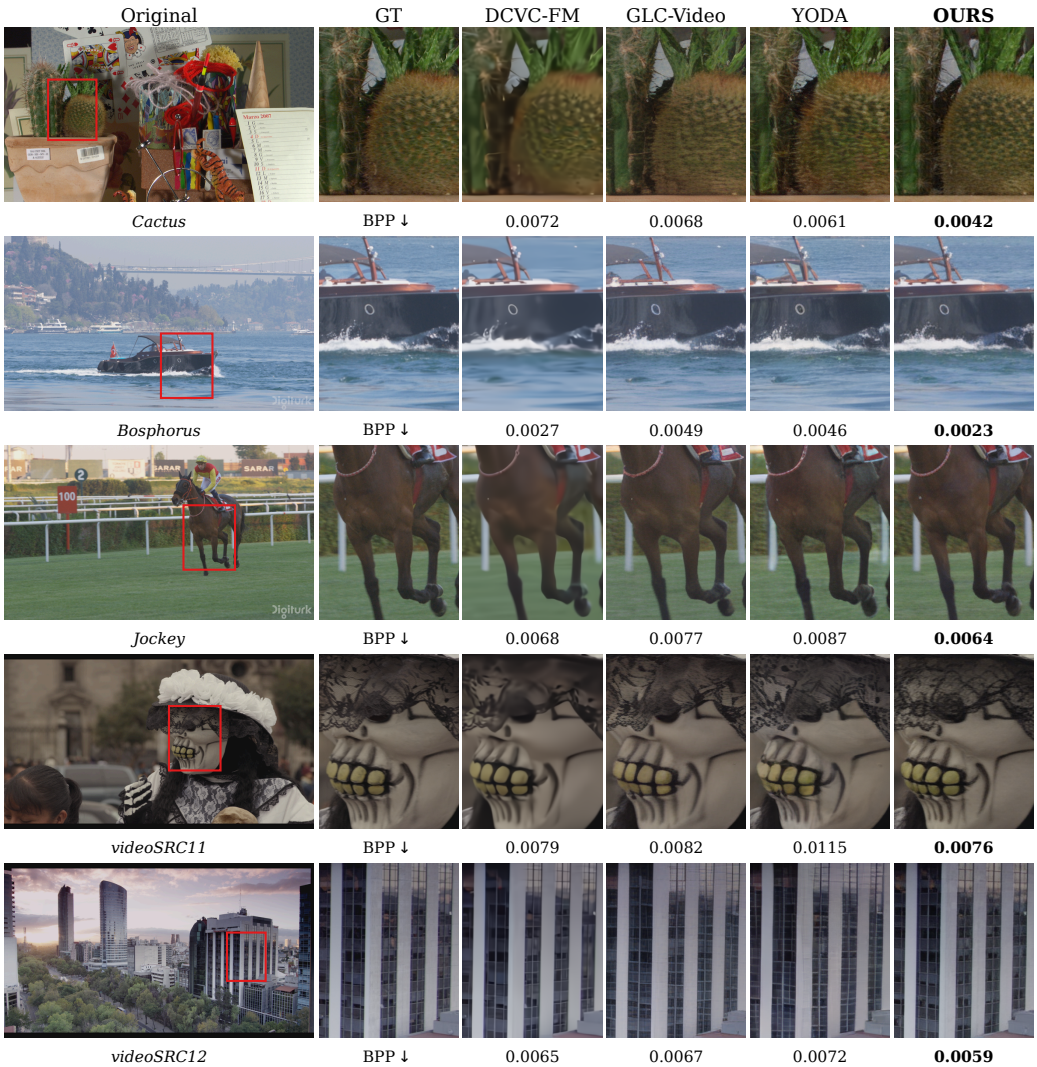


Figure 10: Additional spatial detail comparisons. Zoom in to inspect local textures.