RIFT: RESIDUAL-INFORMED FLOW TRANSFER FOR ONE-STEP GENERATIVE VIDEO COMPRESSION

Anonymous authors
Paper under double-blind review

ABSTRACT

Recent advances in video generation have enabled generative video compression to reconstruct details lost during coding using pretrained spatiotemporal priors. However, a substantial gap remains between codec-induced degradation and the perturbation trajectories learned by flow-matching models. In this paper, we identify a compression-dependent directional mismatch that degradation-strength-based timestep calibration alone cannot resolve. Compressed latents remain strongly aligned across codec operating points, while their reconstruction residuals vary substantially in magnitude and direction. To bridge this gap, we propose **RIFT**, a **R**esidual-**I**nformed **F**low **T**ransfer framework for one-step generative video compression. At its core, RIFT introduces a **Residual Geometry Inducer** (RGI) that converts the pretrained flow prediction into a content-adaptive restoration update using decoder-visible codec cues. RGI integrates flow-timestep conditioning with a learned full-band residual-strength estimate and content-conditioned direction and gain modulation, aligning the orientation and magnitude of the update with the compression-dependent residual geometry. Extensive experiments on three public video compression benchmarks show that RIFT outperforms competing generative video compression methods.

1 INTRODUCTION

Recent advances in video generation have produced pretrained models with powerful spatiotemporal priors (Wan et al., 2025). Generative video compression can exploit these priors to restore visually plausible detail that would be prohibitively expensive to transmit at low bitrates (Mao et al., 2025; Ling et al., 2026). Yet repeatedly evaluating a large generative model makes decoding costly. One-step reconstruction offers a more practical path (Xue et al., 2026a; Li et al., 2026), provided that the generative prior can be adapted to the errors of an actual video codec.

This adaptation is nontrivial because codec-induced artifacts differ from the perturbations encountered during generative pretraining (Relic et al., 2025). Compression-aware conditioning and generative-state selection help bridge this gap (Mao et al., 2025; Ling et al., 2026), but a scalar measure of degradation cannot, on its own, specify the vector-valued correction needed by a decoded latent. In this paper, we revisit the geometry of codec residuals across compression operating points. Although decoded latents at different operating points remain strongly aligned, their clean-minus-decoded residuals change appreciably in both magnitude and direction. This *compression-dependent directional mismatch* means that adjusting a flow timestep or scaling a shared update is insufficient to describe the desired restoration trajectory.

To address this mismatch, we propose **RIFT** (**R**esidual-**I**nformed **F**low **T**ransfer), a one-step generative video compression framework that adapts a pretrained video flow prior to the residuals of entropy-decoded latents. Unlike timestep-only adaptation, RIFT does not assume that codec-dependent errors can be corrected by moving to a different point on an otherwise shared restoration trajectory. RIFT therefore introduces the **Residual Geometry Inducer** (RGI), which turns a single backbone prediction into a content-adaptive correction whose direction and magnitude respond to local codec degradation. RGI combines a learned full-band residual-strength estimate, a flow timestep selected by a fixed operating-point lookup table, and learned direction and gain modulation. We design the lookup table to assign each codec operating point a reproducible backbone condition, while the learned components adjust the orientation and amplitude of its one-step update. RGI

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.

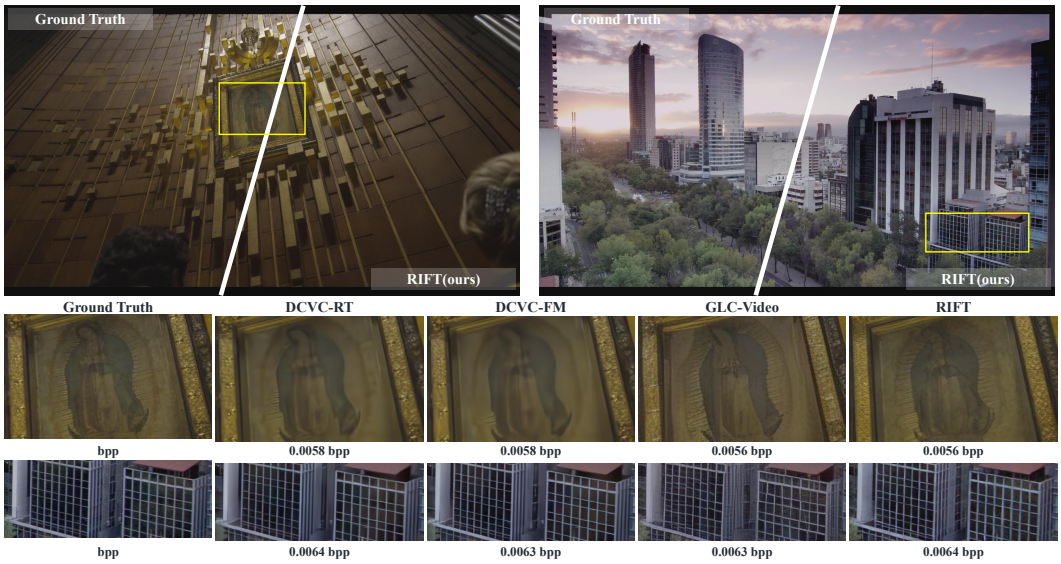

Figure 1: Qualitative comparison at ultra-low bitrates. Conventional and learned codecs produce blurred frames, while generative methods such as GLC-Video recover sharper textures but may introduce structural artifacts and inconsistent details. RIFT bridges the mismatch between compressed latents and the generative DiT through a lightweight codec–flow interface, restoring realistic textures while preserving structural fidelity in a single inference step.

additionally uses decoder-visible spatial codec cues to localize degradation without requiring an additional conditioning payload.

We evaluate RIFT on HEVC-B, MCL-JCV, and UVG (Bossen, 2013; Wang et al., 2016; Mercat et al., 2020) using actual entropy-coded bitstreams. Across these three public video compression benchmarks, RIFT outperforms competing generative video compression methods, as shown in Fig. 1.

Our main contributions are summarized as follows.

- We identify a compression-dependent directional mismatch in which decoded latents remain aligned across codec operating points while their reconstruction residuals differ substantially in direction and magnitude.
- We propose RIFT, a one-step generative video compression framework that directly restores entropy-decoded latents with a pretrained video flow prior adapted to compression-dependent residual geometry.
- We introduce RGI to bridge codec degradation and the pretrained flow trajectory through learned full-band residual-strength estimation, direction refinement, and gain modulation at a reproducible codec-conditioned flow timestep.
- Extensive experiments on HEVC-B, MCL-JCV, and UVG show that RIFT outperforms state-of-the-art generative video compression methods in perceptual quality, with RIFT's rates measured from actual entropy-coded bitstreams.

2 Related Work

2.1 Learned and Generative Video Compression

Learned video compression has progressed from neural predictive coding to context-conditioned representation learning. Early methods integrate motion compensation and residual coding into end-to-end optimization (Lu et al., 2019). Subsequent approaches exploit propagated features, multi-scale temporal contexts, and bidirectional references to improve conditional coding (Li et al., 2021; Sheng et al., 2021; Li et al., 2023; Sheng et al., 2024a;b). Complementary developments improve

entropy estimation, rate adaptation, and practical efficiency through spatial-temporal probability modeling and efficient codec architectures (Li et al., 2022; 2024; Jia et al., 2025). These advances improve rate–distortion performance, while perceptual reconstruction at ultra-low bitrates motivates incorporating generative priors.

Generative compression uses adversarial objectives and learned generative representations to synthesize plausible details from compact bitstreams (Mentzer et al., 2020; Yang et al., 2022; Qi et al., 2025). Diffusion-based approaches further combine pretrained priors with temporal conditioning, semantic guidance, and efficient single-step decoding (Ma & Chen, 2025; Xue et al., 2026a; Li et al., 2026). Video-native priors extend this direction toward sequence-level refinement and trajectory-based compression (Mao et al., 2025; Ling et al., 2026), while quantization-aware diffusion formulations address discrepancies between codec degradation and generative pretraining (Relic et al., 2025). RIFT complements these approaches by explicitly modeling quantization-dependent residual geometry through decoupled timestep conditioning, direction prediction, and gain control.

2.2 Video Generation Models

Video generation has advanced through both discrete-token modeling and continuous diffusion. Token-based approaches use masked or autoregressive transformers to model compressed video sequences conditioned on text and other modalities (Villegas et al., 2022; Kondratyuk et al., 2023). Diffusion-based methods extend image generators with temporal modeling and spatial-temporal upsampling to synthesize coherent, high-resolution videos (Ho et al., 2022b; Singer et al., 2022; Ho et al., 2022a). Latent diffusion further reduces computational demands and facilitates transferring pretrained image-generation capabilities to video (Blattmann et al., 2023b;a).

Building on diffusion transformers (Peebles & Xie, 2023), recent models combine spatiotemporal latent representations with joint or factorized attention to improve video generation (Ma et al., 2024; Yang et al., 2024; Zheng et al., 2024). Large-scale training and flow-matching formulations further support powerful video foundation models with rich appearance and motion priors (Lipman et al., 2022; Kong et al., 2024; Wan et al., 2025). These developments provide a strong foundation for generative reconstruction, but adapting pretrained models to compressed latents requires accounting for codec-specific degradation. RIFT introduces a residual-informed interface around a pretrained video flow backbone, enabling codec-dependent correction within a single model evaluation.

3 Compression-to-Flow Mismatch

Let $\mathbf{z}$ denote the clean video latent and $\hat{\mathbf{z}}_q$ its compressed observation at codec operating point q. Compression produces rate-constrained, content-dependent residuals, whereas a pretrained flow model follows a generative trajectory indexed by timestep τ. Thus q and τ describe different processes, and a flow timestep alone cannot determine the direction and magnitude of $\mathbf{z} - \hat{\mathbf{z}}_q$.

3.1 Compression Changes Residual Geometry

The codec correction is the clean-minus-decoded residual

$$\mathbf{r}_q = \mathbf{z} - \hat{\mathbf{z}}_q. \tag{1}$$

If QP changed only degradation severity along a common trajectory, the residuals would satisfy $\mathbf{r}_q \approx a_q \mathbf{r}_{\mathrm{ref}}$ for some scalar a_q. Measurements from actual codec encode–decode passes contradict this approximation: decoded latents remain strongly aligned across QPs, while their residuals vary substantially in both norm and direction. Detailed measurements are provided in Appendix A.2. Thus, a flow timestep can select a useful operating point for the pretrained prior, but cannot specify the content-dependent correction direction.

This does not rule out nonlinear flow trajectories or timestep-dependent prediction directions. Rather, it shows that a shared-direction, scalar-rescaling model is insufficient for codec residuals. RIFT therefore factorizes the restoration process: the flow timestep selects a stable operating point for the frozen prior, codec cues condition residual strength, and learned corrections adjust the direction and magnitude of the resulting update. These cues are used as conditioning evidence rather than direct observations of the clean residual.

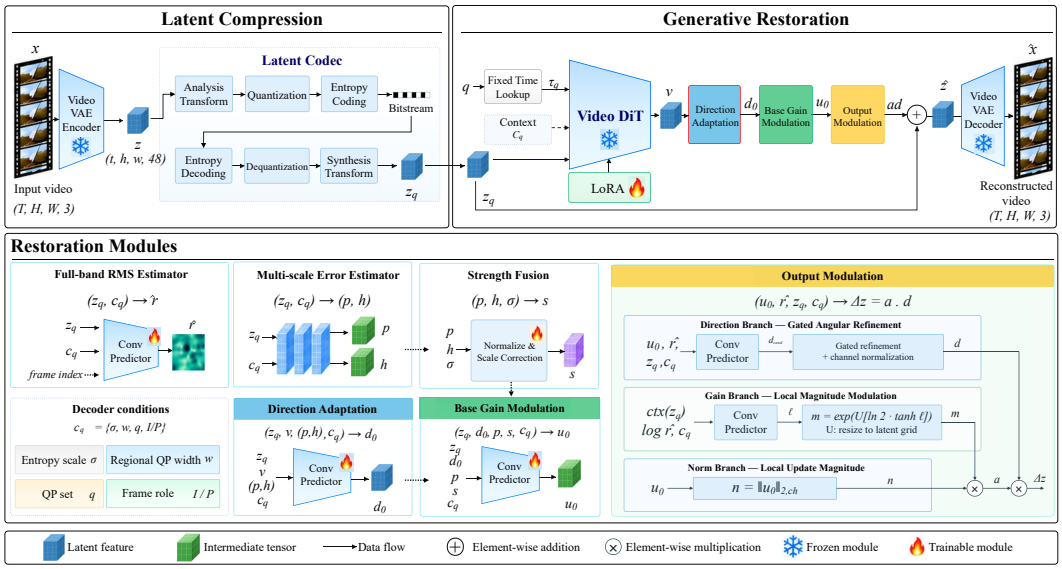

Figure 2: **Overview of RIFT.** The latent codec reconstructs the compressed representation, and RGI adapts a single flow prediction using decoder-visible conditions to restore it in one step.

3.2 Residual Strength Does Not Determine Update Magnitude

Residual strength alone cannot determine a restoration vector. For a predicted update $\mathbf{d}$ and true residual $\mathbf{r}$, the gain minimizing endpoint error is

$$g^* = \frac{\langle \mathbf{r}, \mathbf{d} \rangle}{\|\mathbf{d}\|_2^2}. \tag{2}$$

Hence, even at fixed residual strength, the optimal gain changes when the predicted direction is imperfect. RIFT therefore separates residual strength, direction correction, and gain modulation, while the flow timestep determines the backbone operating point.

These roles form a decoder-compatible interface rather than quantities learned in the same manner. RIFT estimates token-wise residual strength to condition learned corrections, while the QP-to-flow-timestep mapping is specified by a designed lookup table. Direction correction and gain modulation then refine the backbone update, making each signal's role explicit without conflating residual strength with a learned flow timestep.

4 Method

RIFT combines latent compression with one-step generative restoration. A frozen video VAE encoder maps the input video $\mathbf{x}$ to $\mathbf{z} = \mathcal{E}(\mathbf{x})$. The latent codec produces a bitstream and reconstructs $\hat{\mathbf{z}}_q$ at operating point q. The Residual Geometry Inducer (RGI) adapts a pretrained video flow prediction to this decoded observation, producing a restored latent $\tilde{\mathbf{z}}$. The reconstructed video is $\hat{\mathbf{x}} = \mathcal{D}(\tilde{\mathbf{z}})$, where $\mathcal{D}$ is the frozen VAE decoder. Figure 2 summarizes the framework.

4.1 Residual Geometry Inducer

Decoder conditions. RGI uses only the decoded latent and conditions available at the decoder:

$$\mathbf{c}_q = \{\sigma, w, q, \rho\}, \tag{3}$$

where σ is the entropy-model scale, w is the regional quantization width, and ρ denotes the latent I/P role. Normalized temporal position is derived internally.

The regional quantizer is a supporting component of the latent codec. It predicts w from entropy-model context shared by the encoder and decoder, allowing the decoder to reproduce the same

regional widths without transmitting an additional dense map. The regional QP width w and entropy scale σ provide complementary information about local quantization and coding uncertainty.

Full-band RMS Estimator. The Full-band RMS Estimator predicts a spatially and temporally varying residual-strength map:

$$\hat{\mathbf{r}} = \mathcal{R}_{\mathrm{RMS}}\left(\hat{\mathbf{z}}_q, \mathbf{c}_q, \eta\right), \tag{4}$$

where η denotes the normalized frame index. The prediction $\hat{\mathbf{r}}$ is directly supervised by the local RMS magnitude of the clean-minus-decoded latent residual, providing an explicitly calibrated measure of local codec degradation.

Multi-scale Error Estimator. In parallel, the Multi-scale Error Estimator extracts positive degradation descriptors and shared contextual features:

$$(\mathbf{p}, \mathbf{h}) = \mathcal{E}_{\mathrm{err}}\left(\hat{\mathbf{z}}_q, \mathbf{c}_q\right). \tag{5}$$

Here, $\mathbf{p}$ contains positive error descriptors at multiple scales, while $\mathbf{h}$ retains content-dependent contextual information. Unlike the explicitly supervised RMS prediction, these features are learned to support the subsequent restoration modules.

Strength Fusion. The Strength Fusion module combines the multi-scale descriptors with the entropy-model scale:

$$\mathbf{s} = \mathcal{S}\left(\mathbf{p}, \mathbf{h}, \sigma\right). \tag{6}$$

Internally, this module normalizes the predicted descriptors and applies a learned scale correction. The resulting feature $\mathbf{s}$ represents codec-conditioned degradation strength for Base Gain Modulation. The two strength paths therefore serve complementary purposes: $\hat{\mathbf{r}}$ provides explicitly supervised physical calibration, whereas $(\mathbf{p}, \mathbf{h}, \mathbf{s})$ provide contextual features for restoration.

Fixed Time Lookup and one-step Video DiT prediction. The operating point selects a fixed Video DiT timestep through the Fixed Time Lookup:

$$\tau_q = \begin{cases} 520, & q = 0, \\ 640, & q = 16, \\ 800, & q = 39, \\ 960, & q = 63. \end{cases} \tag{7}$$

Using the decoded latent, the selected timestep, and the embedded decoder context, RGI queries the Video DiT once:

$$\mathbf{v} = \mathcal{F}_{\phi,\theta_{\mathrm{L}}}\left(\hat{\mathbf{z}}_q, \tau_q; \mathbf{c}_q\right), \tag{8}$$

where ϕ denotes the fixed foundation-model weights and θ_{L} denotes the trainable LoRA parameters. Inference introduces no additional noise and requires neither iterative denoising nor numerical flow integration.

Geometry adaptation and output modulation. The raw flow prediction $\mathbf{v}$ is first adapted to the codec-dependent residual geometry. Decoder Adaptation predicts a correction using the decoded latent, multi-scale error features, and decoder conditions, while Base Gain Modulation converts the adapted direction into a restoration proposal:

$$\mathbf{d}_0 = \mathbf{v} + \mathcal{A}\left(\hat{\mathbf{z}}_q, \mathbf{v}, \mathbf{p}, \mathbf{h}, \mathbf{c}_q\right), \ \mathbf{g}_{\mathrm{base}} \quad = \mathcal{G}_{\mathrm{base}}\left(\hat{\mathbf{z}}_q, \mathbf{d}_0, \mathbf{p}, \mathbf{s}, \mathbf{c}_q\right), \ \mathbf{u}_0 = \mathbf{g}_{\mathrm{base}} \odot \mathbf{d}_0. \tag{9}$$

Output Modulation then decomposes $\mathbf{u}_0$ into a normalized direction and a local magnitude. The Norm Branch extracts

$$\mathbf{n} = \left\|\mathbf{u}_0\right\|_{2,\mathrm{ch}}, \qquad \mathbf{e}_0 = \mathcal{N}(\mathbf{u}_0). \tag{10}$$

where $\mathcal{N}$ denotes channel-wise normalization at each spatiotemporal latent position. The Direction Branch predicts a candidate direction and applies a frame-role-dependent angular refinement:

$$\mathbf{d}_{\mathrm{cand}} = \mathcal{G}_{\mathrm{dir}}\left(\mathbf{u}_0, \hat{\mathbf{r}}, \hat{\mathbf{z}}_q, \mathbf{c}_q\right), \ \mathbf{d} \quad = \mathcal{N}\left(\mathbf{e}_0 + \beta_\rho\left(\mathbf{d}_{\mathrm{cand}} - \mathbf{e}_0\right)\right). \tag{11}$$

In parallel, the Gain Branch predicts a bounded local adjustment:

$$\boldsymbol{\ell} = \mathcal{G}_{\mathrm{out}}(\mathrm{ctx}(\hat{\mathbf{z}}_q), \log \mathrm{sg}(\hat{\mathbf{r}}), \mathbf{c}_q), \quad \mathbf{m} = \exp(\mathcal{U}[\log 2 \cdot \tanh(\boldsymbol{\ell})]), \quad \mathbf{a} = \mathbf{n} \odot \mathbf{m}. \tag{12}$$

where sg denotes stop-gradient and $\mathcal{U}$ resizes the prediction to the latent grid. This parameterization constrains $\mathbf{m}$ to $[1/2, 2]$.

The final latent correction and restored latent are

$$\Delta\mathbf{z} = \mathbf{a} \odot \mathbf{d}, \qquad \tilde{\mathbf{z}} = \hat{\mathbf{z}}_q + \Delta\mathbf{z}. \tag{13}$$

This decomposition allows RGI to refine the restoration direction separately from its local magnitude.

4.2 Training Objectives

Training is divided into two curricula that separately optimize the latent codec and the generative restoration pathway.

Codec curriculum. The codec curriculum optimizes the transmitted latent representation using a combination of latent reconstruction and perceptual distortion:

$$\mathcal{D}_{\text{codec}} = \mathcal{L}_{\text{latent}} + \mathcal{L}_{\text{LPIPS}}, \tag{14}$$

where $\mathcal{L}_{\text{latent}}$ is the Smooth L1 distance between the decoded latent $\hat{\mathbf{z}}_q$ and the clean latent $\mathbf{z}$. The corresponding objective is

$$\mathcal{L}_{\text{codec}} = R + \lambda_q \mathcal{D}_{\text{codec}}, \tag{15}$$

where R is the differentiable entropy-rate estimate.

Generative restoration curriculum. The generative restoration curriculum optimizes the codec–flow pathway for perceptual reconstruction. Let $\hat{\mathbf{x}} = \mathcal{D}(\tilde{\mathbf{z}})$ denote the restored video. Its distortion objective is

$$\mathcal{D}_{\text{rest}} = 0.2\,\mathcal{L}_{\text{RGB}} + \mathcal{L}_{\text{LPIPS}} + 0.1\,\mathcal{L}_{\text{DISTS}} + \mathcal{L}_{\text{latent}}, \tag{16}$$

where $\mathcal{L}_{\text{RGB}}$ measures RGB L1 distortion, while $\mathcal{L}_{\text{LPIPS}}$ and $\mathcal{L}_{\text{DISTS}}$ provide complementary perceptual supervision. The corresponding objective is

$$\mathcal{L}_{\text{rest}} = R + \lambda_q \mathcal{D}_{\text{rest}}. \tag{17}$$

Both curricula use the same rate–distortion weights: $\lambda_q \in \{0.15, 0.08, 0.05, 0.02\}$ for $q \in \{0, 16, 39, 63\}$, respectively. Quantization is approximated by a straight-through estimator during training, whereas all reported rates are measured from actual rANS-coded bitstreams.

5 Experiments

5.1 Experimental Setup

Datasets and protocol. We evaluate on HEVC Class B (Bossen, 2013), MCL-JCV (Wang et al., 2016), and UVG (Mercat et al., 2020), comprising 42 sequences. For each sequence, we code the first 96 frames in four independently decoded groups of 25, 25, 25, and 21 frames. We evaluate QPs $\{0, 16, 39, 63\}$ at both native 1920×1080 resolution and 1280×720, obtained by resizing the source RGB frames.

Implementation. We first train the conditional latent codec on windows from Vimeo90K (Xue et al., 2019) and OpenVidHD (Nan et al., 2025). We then adapt the pretrained Wan video flow prior with rank-8 LoRA and the decoder-visible RGI conditions described in Section 4.1. Training is conducted on six NVIDIA H800 GPUs, with batch sizes of 3 and 1 for 256×256 and 512×512 training crops, respectively. All reported RIFT reconstructions are obtained from independently entropy-decoded rANS bitstreams rather than from a differentiable quantization surrogate.

Metrics and comparisons. We compute rate from the actual coded payload and normalize it by the total RGB pixel count of the encoded canvas, including codec padding. We report RGB PSNR, LPIPS-VGG (Zhang et al., 2018), DISTS (Ding et al., 2020), and patch-based FID. FID uses features extracted from 256×256 patches, with statistics pooled across sequences within each dataset. Codec-only reconstruction from the same bitstream isolates the effect of RGI. GLC-Video (Qi et al., 2025) and GNVC-VD (Mao et al., 2025) serve as the main external generative references. Cross-method rate–quality comparisons use matched frame ranges, resolutions, and metric definitions, and are restricted to overlapping bitrate ranges. Same-QP comparisons are used only for diagnosis.

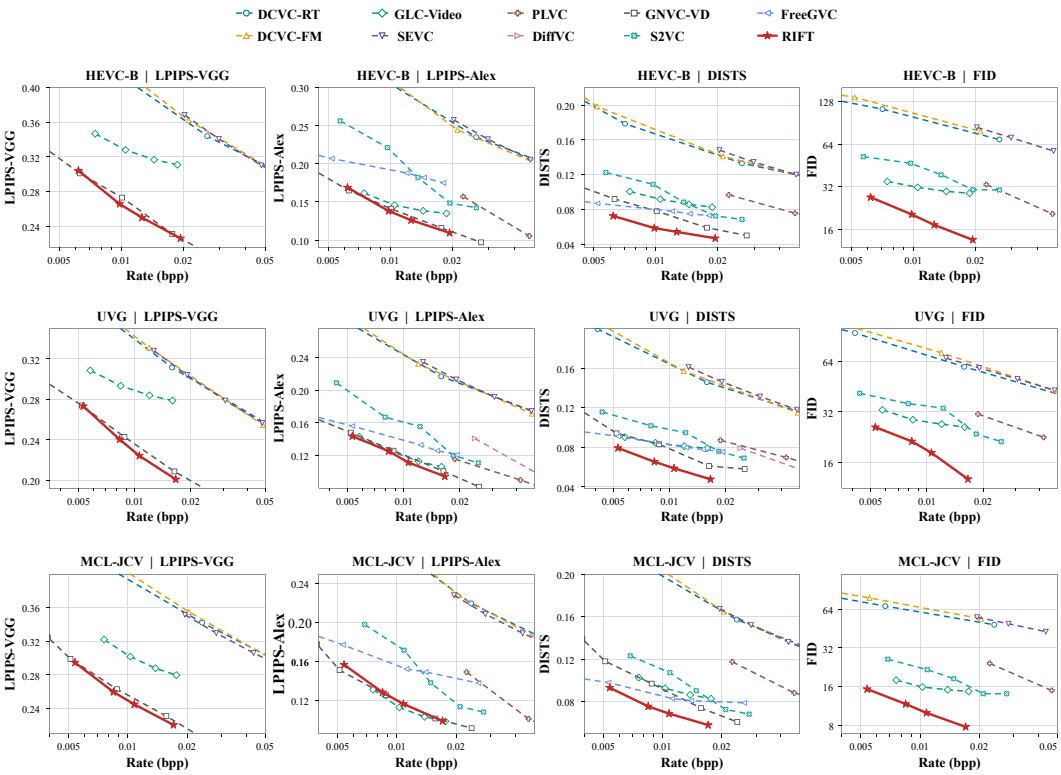

Figure 3: Rate–quality comparisons at 1080p on HEVC Class B, UVG, and MCL-JCV.

5.2 Main Results

Fig. 3 and Fig. 8 compare RIFT with DCVC-RT (Jia et al., 2025), DCVC-FM (Li et al., 2024), SEVC (Bian et al., 2025), PLVC (Yang et al., 2022), DiffVC (Ma & Chen, 2025), S2VC (Xue et al., 2026b), FreeGVC (Ling et al., 2026), GLC-Video (Qi et al., 2025), and GNVC-VD (Mao et al., 2025) at 1080p and 720p, respectively. We locally reproduce and evaluate DCVC-RT, DCVC-FM, and GLC-Video, whereas the results for the remaining baselines are taken from their corresponding publications. For GLC-Video, we follow its original inference protocol, which conditionally codes each latent using the previously decoded latent as temporal context and directly reconstructs each frame without iterative diffusion refinement. The first 96 frames are evaluated with the intra-period set to -1. For GNVC-VD, we directly use the results reported in its original paper, where each 96-frame sequence is divided into four groups of $25/25/25/21$ frames and reconstructed using five flow-refinement steps.

Across the three datasets, RIFT achieves lower DISTS, LPIPS-VGG, and FID than GLC-Video over their overlapping bitrate ranges. Its advantage in DISTS remains consistent across both resolutions and multiple operating points.

5.3 Qualitative Results

Fig. 4 compares RIFT with DCVC-RT, DCVC-FM, and GLC-Video at closely matched bitrates. RIFT retains more distinct surface patterns in the water scene and clearer thin boundaries and repeated structures in the architectural scene. The facial and vegetation examples likewise show recognizable contours and fine appearance variations. Compared with DCVC-RT and DCVC-FM, RIFT produces less smooth reconstructions; compared with GLC-Video, its displayed details agree more closely with the reference.

Ablation. We assess the dependence of the RIFT decoder on three restoration components. BD-rate is measured relative to RIFT over the common quality interval of dataset-averaged curves.

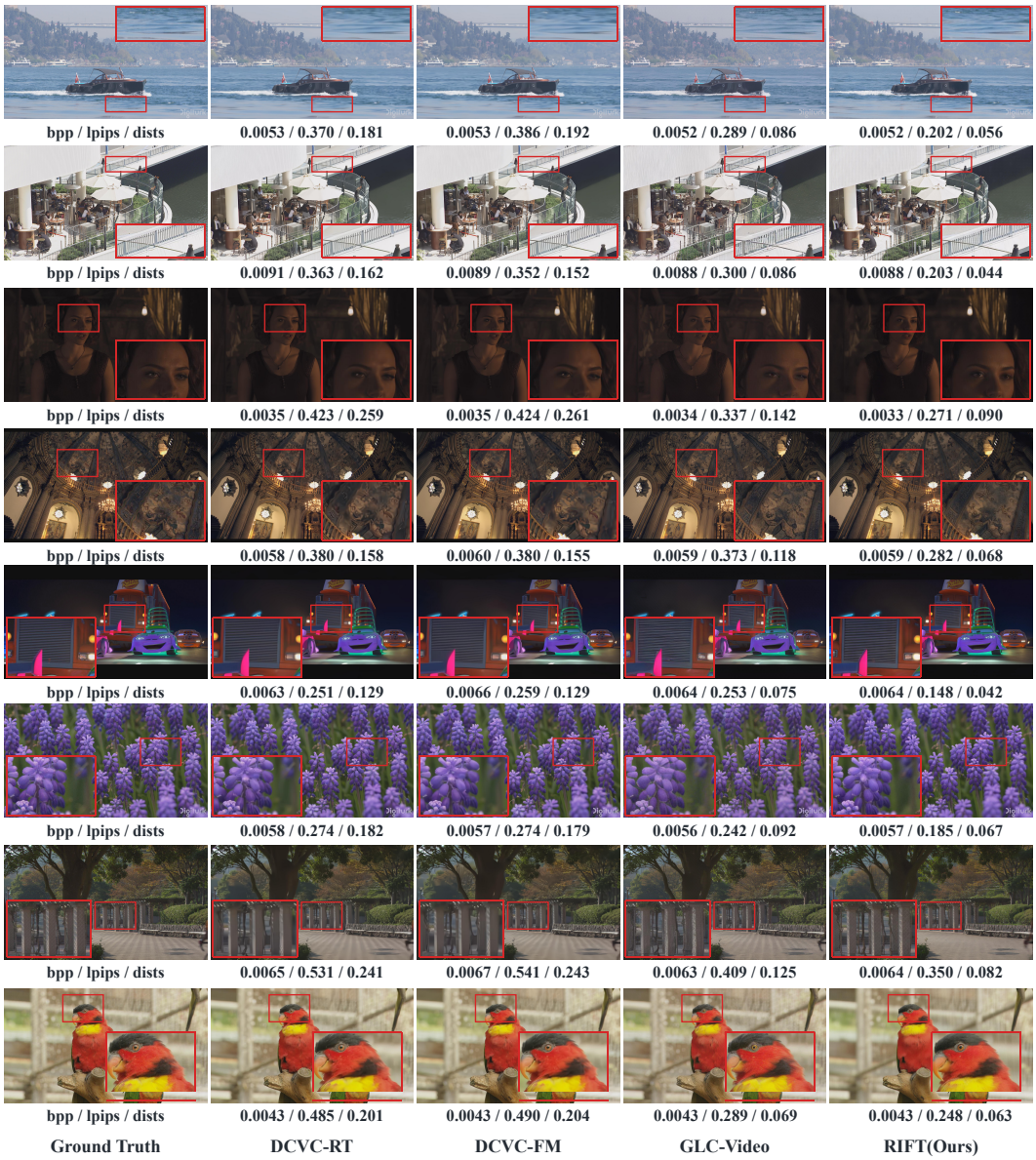

Figure 4: Qualitative comparison on eight sequences from HEVC Class B, UVG, and MCL-JCV. Red boxes indicate regions shown in enlarged insets.

For the multi-scale error ablation, predicted errors are replaced by the stored error reference and shared features are set to zero, while scale correction and strength fusion remain active. For the direction ablation, both the inherited direction residual and the subsequent angular refinement are bypassed. For the gain ablation, learned base, local, and global gain corrections are bypassed, while the fixed QP gain table is retained and the final learned multiplier is set to one. Each intervention is applied separately.

As shown in Table 1, removing learned gain modulation produces the largest penalty among the three module interventions: 20.54% BD-rate on LPIPS and 24.30% on DISTS. Bypassing multi-scale error estimation incurs penalties of 7.48% and 9.76%, while bypassing direction refinement incurs 0.96% and 3.76%, respectively. These results show that learned magnitude modulation has the strongest effect among these inference-time interventions.

Table 1: Ablation on test sequences at 1080p. BD-rate is measured relative to full RIFT; positive values indicate increased rate at equal perceptual quality.

Configuration	BD-rate (LPIPS) (%)	BD-rate (DISTS) (%)
w/o Multi-scale Error	+7.48	+9.76
w/o Direction Refinement	+0.96	+3.76
w/o Learned Gain	+20.54	+24.30

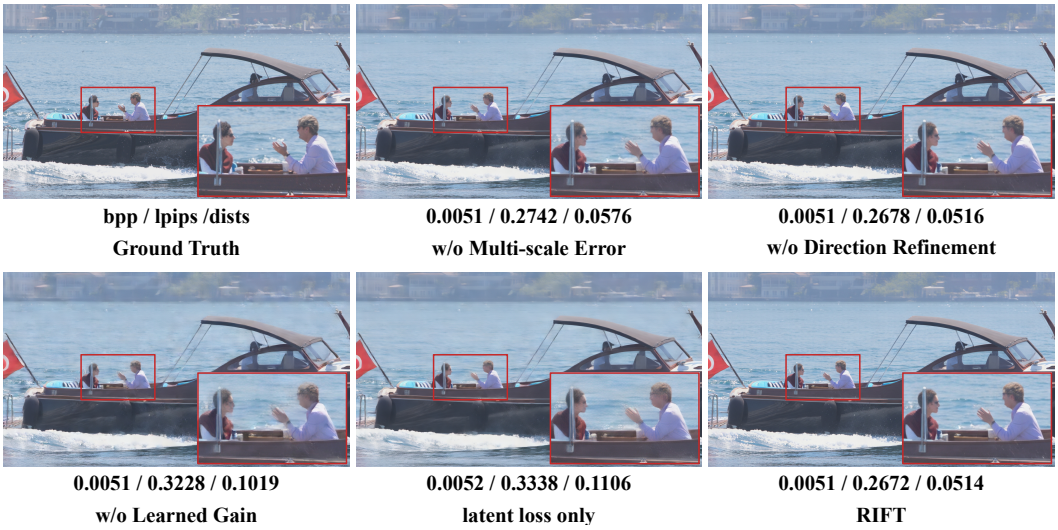

Figure 5: Qualitative comparison of the ablation variants. All variants are evaluated on the same frame and spatial crop.

We further evaluate a variant trained without the RGB-domain reconstruction and perceptual losses, including RGB, LPIPS, and DISTS, while retaining only latent-domain supervision. This variant yields BD-rate increases of 50.95% in LPIPS and 91.58% in DISTS relative to the full model, indicating substantial performance degradation. These results show that latent-domain supervision alone is insufficient to effectively guide generative restoration, whereas image-domain reconstruction and perceptual objectives are essential for improving the quality of the final decoded video.

Figure 5 shows qualitative examples with identical frames and crops across methods. The three module interventions use the same RIFT bitstream, whereas the latent-only variant is trained separately and its bitrate is measured from its own entropy-coded bitstream. Additional codec-only comparisons are provided in Appendix D.

6 Conclusion

We present **RIFT**, a Residual-Informed Flow Transfer framework for one-step generative video compression. Our analysis shows that a scalar flow time alone is insufficient to describe codec degradation, motivating the explicit separation of residual strength, flow time, correction direction, and gain. RIFT combines this factorized interface with rank-8 LoRA adaptation and one-step reconstruction, enabling efficient use of a video foundation model for compression restoration. Experimental results demonstrate the effectiveness of modeling codec residuals beyond scalar timestep alignment and show that RIFT outperforms competing generative video compression methods.

AI Use Statement

Generative AI tools were used to assist with literature retrieval, manuscript organization and language refinement, code implementation, experimental design, discussion of the conceptual and mathemat-

ical formulation, interpretation and presentation of diagnostic analyses, and the design and visual refinement of manuscript figures. All AI-assisted code was inspected and tested by the authors, and all experimental designs and methodological decisions were independently reviewed and approved by the authors. All experimental data and numerical measurements reported in this paper were obtained from actual codec training and evaluation runs rather than generated by AI tools. The authors verified the experimental results, mathematical claims, bibliographic information, and all AI-assisted text, code, and figures, and take full responsibility for the final content and conclusions of this work.

Reproducibility Statement

The codec–flow interface, training objectives, and inference procedure are described in Sections 4.1 and 4.2. The complete training curriculum, including the datasets, iteration budgets, optimized parameter groups, optimizer, and learning-rate schedules, is documented in Appendix B and Tables 4 and 5. Additional analyses of residual distributions and cross-operating-point geometry are provided in Appendix A. The supplementary material includes the implementation details, evaluation configurations, and code required to reproduce the reported experiments.

References

Yifan Bian, Chuanbo Tang, Li Li, and Dong Liu. Augmented deep contexts for spatially embedded video coding. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2094–2104. IEEE, 2025.

Andreas Blattmann, Tim Dockhorn, Sumith Kulal, Daniel Mendelevitch, Maciej Kilian, Dominik Lorenz, Yam Levi, Zion English, Vikram Voleti, Adam Letts, Varun Jampani, and Robin Rombach. Stable video diffusion: Scaling latent video diffusion models to large datasets. *arXiv preprint arXiv:2311.15127*, 2023a. URL https://arxiv.org/abs/2311.15127.

Andreas Blattmann, Robin Rombach, Huan Ling, Tim Dockhorn, Seung Wook Kim, Sanja Fidler, and Karsten Kreis. Align your latents: High-resolution video synthesis with latent diffusion models. *arXiv preprint arXiv:2304.08818*, 2023b. URL https://arxiv.org/abs/2304.08818.

Frank Bossen. Common test conditions and software reference configurations. Technical Report JCTVC-L1100, Joint Collaborative Team on Video Coding (JCT-VC) of ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11, Geneva, Switzerland, 2013.

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*, 2020.

Jonathan Ho, William Chan, Chitwan Saharia, Jay Whang, Ruiqi Gao, Alexey Gritsenko, Diederik P. Kingma, Ben Poole, Mohammad Norouzi, David J. Fleet, and Tim Salimans. Imagen Video: High definition video generation with diffusion models. *arXiv preprint arXiv:2210.02303*, 2022a. URL https://arxiv.org/abs/2210.02303.

Jonathan Ho, Tim Salimans, Alexey Gritsenko, William Chan, Mohammad Norouzi, and David J. Fleet. Video diffusion models. *arXiv preprint arXiv:2204.03458*, 2022b. URL https://arxiv.org/abs/2204.03458.

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.

Dan Kondratyuk, Lijun Yu, Xiuye Gu, José Lezama, Jonathan Huang, Grant Schindler, Rachel Hornung, Vighnesh Birodkar, et al. VideoPoet: A large language model for zero-shot video generation. *arXiv preprint arXiv:2312.14125*, 2023. URL https://arxiv.org/abs/2312.14125.

Weijie Kong, Qi Tian, Zijian Zhang, Rox Min, Zuozhuo Dai, Jin Zhou, Jiangfeng Xiong, Xin Li, et al. HunyuanVideo: A systematic framework for large video generative models. *arXiv preprint arXiv:2412.03603*, 2024. URL https://arxiv.org/abs/2412.03603.

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. Hybrid spatial-temporal entropy modelling for neural video compression. *arXiv preprint arXiv:2207.05894*, 2022. URL https://arxiv.org/abs/2207.05894.

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. doi: 10.1109/CVPR52733.2024.02466.

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*, 2026.

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.

Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. *arXiv preprint arXiv:2210.02747*, 2022. URL https://arxiv.org/abs/2210.02747.

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*, 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.

Xin Ma, Yaohui Wang, Xinyuan Chen, Gengyun Jia, Ziwei Liu, Yuan-Fang Li, Cunjian Chen, and Yu Qiao. Latte: Latent diffusion transformer for video generation. *arXiv preprint arXiv:2401.03048*, 2024. URL https://arxiv.org/abs/2401.03048.

Qi Mao, Hao Cheng, Tinghan Yang, Libiao Jin, and Siwei Ma. Generative neural video compression via video diffusion prior. *arXiv preprint arXiv:2512.05016*, 2025. URL https://arxiv.org/abs/2512.05016.

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

Kepan Nan, Rui Xie, Penghao Zhou, Tiehan Fan, Zhenheng Yang, Zhijie Chen, Xiang Li, Jian Yang, and Ying Tai. Openvid-1m: A large-scale high-quality dataset for text-to-video generation. In *International conference on learning representations*, volume 2025, pp. 1045–1064, 2025.

William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 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.

Lucas Relic, Roberto Azevedo, Yang Zhang, Markus Gross, and Christopher Schroers. Bridging the gap between gaussian diffusion models and universal quantization for image compression. *arXiv preprint arXiv:2504.02579*, 2025. URL https://arxiv.org/abs/2504.02579.

Xihua Sheng, Jiahao Li, Bin Li, Li Li, Dong Liu, and Yan Lu. Temporal context mining for learned video compression. *arXiv preprint arXiv:2111.13850*, 2021. URL https://arxiv.org/abs/2111.13850.

Xihua Sheng, Li Li, Dong Liu, and Houqiang Li. Spatial decomposition and temporal fusion based inter prediction for learned video compression. *arXiv preprint arXiv:2401.15864*, 2024a. URL https://arxiv.org/abs/2401.15864.

Xihua Sheng, Li Li, Dong Liu, and Shiqi Wang. Bi-directional deep contextual video compression. *arXiv preprint arXiv:2408.08604*, 2024b. URL https://arxiv.org/abs/2408.08604.

Uriel Singer, Adam Polyak, Thomas Hayes, Xi Yin, Jie An, Songyang Zhang, Qiyuan Hu, Harry Yang, Oron Ashual, Oran Gafni, Devi Parikh, Sonal Gupta, and Yaniv Taigman. Make-A-Video: Text-to-video generation without text-video data. *arXiv preprint arXiv:2209.14792*, 2022. URL https://arxiv.org/abs/2209.14792.

Ruben Villegas, Mohammad Babaeizadeh, Pieter-Jan Kindermans, Hernan Moraldo, Han Zhang, Mohammad Taghi Saffar, Santiago Castro, Julius Kunze, and Dumitru Erhan. Phenaki: Variable length video generation from open domain textual description. *arXiv preprint arXiv:2210.02399*, 2022. URL https://arxiv.org/abs/2210.02399.

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

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, 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.

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.

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, 2022.

Zhuoyi Yang, Jiayan Teng, Wendi Zheng, Ming Ding, Shiyu Huang, Jiazheng Xu, Yuanming Yang, Wenyi Hong, Xiaohan Zhang, Guanyu Feng, Da Yin, Xiaotao Gu, Yuxuan Zhang, Weihan Wang, Yean Cheng, Ting Liu, Bin Xu, Yuxiao Dong, and Jie Tang. CogVideoX: Text-to-video diffusion models with an expert transformer. *arXiv preprint arXiv:2408.06072*, 2024. URL https://arxiv.org/abs/2408.06072.

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.

Zangwei Zheng, Xiangyu Peng, Tianji Yang, Chenhui Shen, Shenggui Li, Hongxin Liu, Yukun Zhou, Tianyi Li, and Yang You. Open-Sora: Democratizing efficient video production for all. *arXiv preprint arXiv:2412.20404*, 2024. URL https://arxiv.org/abs/2412.20404.

Table 2: Mean KS distances over eight QP/frame-role groups. Gaussian parameters are fitted to the residuals; uniform bounds follow the quantization scale. "Wins" denotes the number of groups for which the Gaussian has a lower KS distance.

Resolution	Residual	Gaussian KS	Uniform KS	Wins
256×256	Raw	0.0273	0.3465	8/8
256×256	Width-normalized	0.0272	0.3481	8/8
512×512	Raw	0.0306	0.3431	8/8
512×512	Width-normalized	0.0309	0.3426	8/8

A Additional Analysis of Codec Residuals

We analyze decoded-latent residuals from actual codec encode–decode passes. These diagnostics support Section 3 and do not measure RGB reconstruction quality.

A.1 Decoded-Latent Residual Distribution

Protocol. For a clean latent $\mathbf{z}$ and its decoded counterpart $\hat{\mathbf{z}}_q$ at QP q, we analyze the codec residual $\mathbf{r}_q$ defined in Eq. (1). At each input resolution (256×256 and 512×512), we use 32 fixed latent windows and QPs $\{0, 16, 39, 63\}$, separating I- and P-frame positions. Each QP/frame-role group contributes up to $300{,}000$ scalar samples; channel-wise analysis uses up to $20{,}000$ samples per channel across 48 channels.

We compare each empirical distribution with a maximum-likelihood Gaussian fit and a uniform reference whose bounds are derived from the codec quantization scale. We quantify their discrepancy using the Kolmogorov–Smirnov (KS) distance,

$$D_n = \sup_x |F_n(x) - F_{\mathrm{ref}}(x)|\,, \tag{18}$$

where F_n and F_{ref} denote the empirical and reference CDFs. We use the KS statistic only as a descriptive measure of distributional mismatch, rather than as a hypothesis test. We report it for both the raw residuals and the residuals normalized by decoder-visible quantization widths.

Results. At both resolutions, the fitted Gaussian has a lower KS distance than the scale-derived uniform reference in all eight QP/frame-role groups, with and without width normalization (Table 2). The channel-wise comparison also favors the Gaussian across all 48 channels at each resolution (Figure 6).

Limits. Residuals approximate a Gaussian but deviate measurably: skewness ranges from 0.09 to 0.24, and excess kurtosis from 0.64 to 1.32. These are residuals after complete codec decoding, not isolated scalar quantization errors. Marginal fit does not establish independence, spatial whiteness, isotropy, or restoration direction. It can inform degradation-strength estimation, while cross-QP directions require separate analysis.

A.2 Cross-QP Geometry of Codec Residuals

Protocol. At each resolution, we encode and decode the same 512 clean latent windows independently at all four QPs. We compare the compressed latents $\hat{\mathbf{z}}_q$ and residuals $\mathbf{r}_q$ defined in Eq. (1) as window vectors and corresponding token vectors. For nonzero vectors,

$$c(\mathbf{a}, \mathbf{b}) = \frac{\langle \mathbf{a}, \mathbf{b} \rangle}{\|\mathbf{a}\|_2 \|\mathbf{b}\|_2}, \tag{19}$$

and the corresponding angle is

$$\theta(\mathbf{a}, \mathbf{b}) = \arccos c(\mathbf{a}, \mathbf{b}). \tag{20}$$

Latent cosine measures representation alignment; residual cosine measures alignment of corrections toward the common clean target.

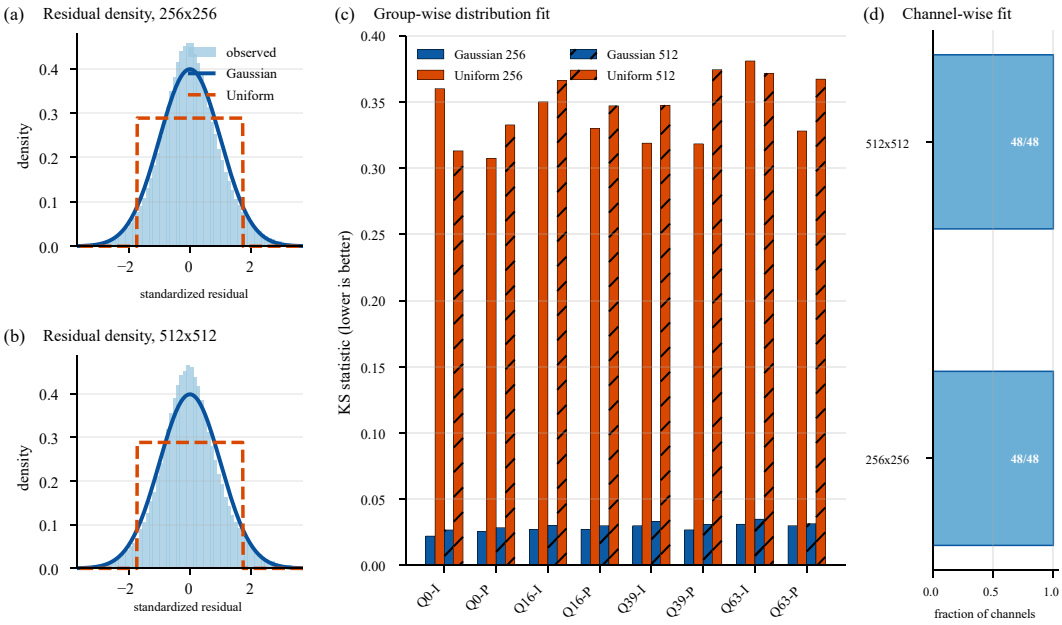

Figure 6: Decoded-latent residual distributions. (a–b) Standardized empirical densities with fitted Gaussian and scale-derived uniform references at 256×256 and 512×512. (c) KS distances by QP and frame role. (d) Fraction of channels favoring the Gaussian.

Table 3: Cross-QP cosine similarity and mean angle. Global and token cosine values average off-diagonal pairs at the window and corresponding-token levels, respectively; angles are measured at the window level.

Resolution	Representation	Global cosine	Mean angle	Token cosine
256×256	Compressed latent	0.9412	19.57°	0.9340
256×256	Residual	0.7080	44.76°	0.7056
512×512	Compressed latent	0.9496	18.08°	0.9419
512×512	Residual	0.7140	44.25°	0.7195

Results. Mean cross-QP cosine similarity exceeds 0.94 for compressed latents but is about 0.71 for residuals, corresponding to mean residual angles near 44° (Table 3). For the $q = 0$ and $q = 63$ pair, latent cosine values are 0.9236 and 0.9331 at the two resolutions, whereas residual cosine is about 0.635 and the angle is near 50.5° (Figure 7). Shared latent content therefore does not imply aligned corrections.

B Training Curriculum

The training curriculum comprises codec representation learning and generative restoration adaptation. It progresses from temporal compression pretraining to interface warm-up, LoRA adaptation, joint optimization, and high-resolution refinement. Tables 4 and 5 specify the training budgets and nominal learning rates.

Codec curriculum. The codec first learns temporal compression from seven-frame RGB sequences in Vimeo-90K for 30K iterations. It subsequently undergoes multi-rate adaptation on OpenVidHD at 256×256 resolution for 20K iterations. OpenVidHD training uses windows containing seven temporal positions in the VAE latent representation, denoted by $T_{\text{latent}} = 7$.

Generative restoration curriculum. We first calibrate the operating-point-to-time mapping using a calibration subset of the training data. The context injection modules and connection heads then

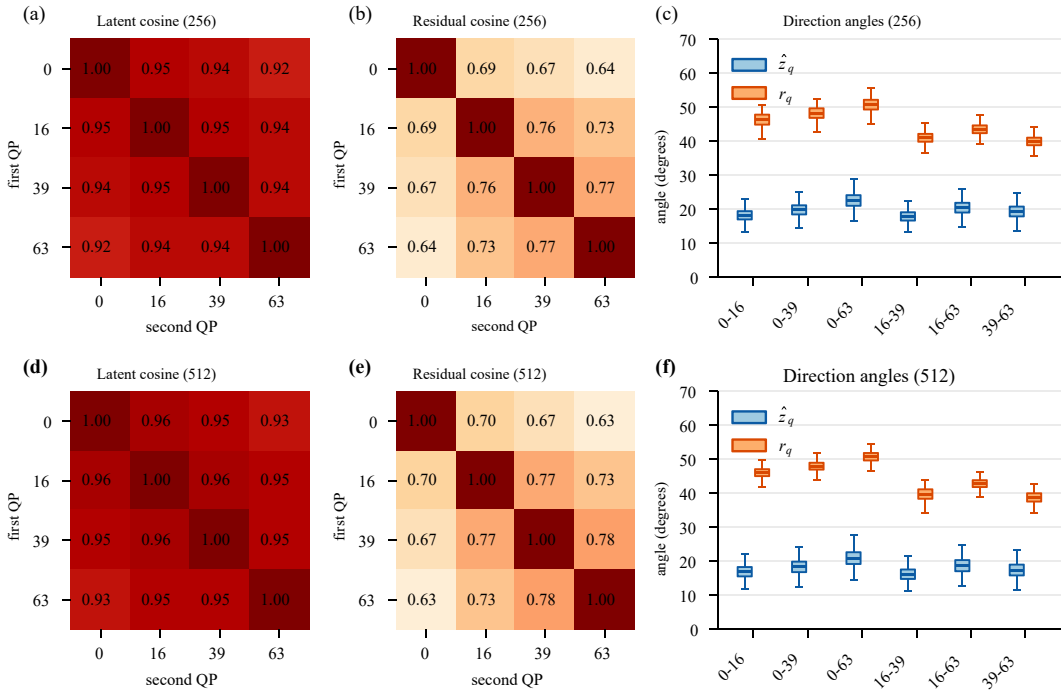

Figure 7: Cross-QP geometry at 256×256 (a–c) and 512×512 (d–f). Columns show compressed-latent cosine similarity, residual cosine similarity, and pairwise direction angles. Each resolution uses 512 matched latent windows.

Table 4: Codec learning curriculum. Learning rates denote nominal values before scheduling.

Stage	Training data	Iterations	Trainable modules and learning rates
Compression pretraining	Vimeo-90K, seven RGB frames	30K	Latent codec: 10^{-4}
Multi-rate adaptation	OpenVidHD, 256×256, $T_{\text{latent}} = 7$	20K	Codec: 3×10^{-6}; regional quantizer: 3×10^{-5}

undergo a short warm-up with the codec and DiT backbone frozen. LoRA adaptation subsequently enables the generative prior to learn compression-residual restoration while the codec provides a stable input distribution.

The joint optimization stage updates the codec together with the restoration modules, allowing the compressed representation and generative decoder to adapt to each other. Training finally progresses to 512×512 windows with lower learning rates to refine high-resolution reconstruction. The VAE and base DiT weights remain frozen throughout; DiT adaptation uses LoRA.

Optimization and supervision. Optimization uses AdamW with linear warm-up followed by cosine learning-rate decay. The strength estimator receives local full-band residual RMS supervision, while the direction and gain branches control the orientation and magnitude of the restoration update. Restoration training combines latent-space, RGB, and perceptual objectives. Joint optimization additionally includes the codec rate–distortion objective. At inference, the fixed operating-point-to-time lookup conditions a single DiT evaluation per latent window.

Table 5: Generative restoration curriculum. Time calibration involves no gradient updates. Context denotes context injection and decoder adaptation modules; heads denote restoration connection heads.

Stage	Training data	Iterations	Trainable modules and learning rates
Time calibration	OpenVidHD calibration subset, 256×256, $T_{\text{latent}} = 7$	—	No parameter updates
Interface warm-up	OpenVidHD, 256×256, $T_{\text{latent}} = 7$	1K	Heads and context: 10^{-4}
LoRA adaptation	OpenVidHD, 256×256, $T_{\text{latent}} = 7$	30K	LoRA, heads, and context: 10^{-4}
Joint optimization	OpenVidHD, 256×256, $T_{\text{latent}} = 7$	30K	Codec: 2×10^{-6}; quantizer: 10^{-5}; strength/gain heads: 3×10^{-5}; direction head: 10^{-5}; inherited heads/context: 5×10^{-6}; LoRA: 10^{-5}
High-resolution refinement	OpenVidHD, 512×512, $T_{\text{latent}} = 7$	8K	Codec: 10^{-6}; quantizer: 5×10^{-6}; strength/gain heads: 2×10^{-5}; direction head: 10^{-5}; inherited heads/context: 3×10^{-6}; LoRA: 5×10^{-6}

C Additional Quantitative Results

C.1 Results at 720p

Figure 8 presents the rate–quality results at 720p on HEVC-B, UVG, and MCL-JCV. For this comparison, we locally reproduce DCVC-RT, DCVC-FM, and GLC-Video and evaluate them together with RIFT under the same test setting. The two distortion-oriented codecs, DCVC-RT and DCVC-FM, achieve substantially higher PSNR, but their perceptual metrics deteriorate rapidly toward the low-rate regime. This behavior reflects the expected distortion–perception trade-off: optimizing pixel-wise fidelity tends to produce smooth reconstructions when the bitrate becomes insufficient to preserve fine visual details.

Compared with GLC-Video over their overlapping bitrate ranges, RIFT consistently achieves lower LPIPS-VGG, LPIPS-Alex, DISTS, and FID across the three datasets. The improvements are particularly clear for DISTS and FID, indicating better preservation of perceptually important structures and more realistic reconstructed-frame statistics. RIFT also provides higher PSNR than GLC-Video throughout the common rate range. Although its PSNR remains below that of the distortion-oriented codecs, the simultaneous improvement over GLC-Video in both pixel fidelity and perceptual quality suggests that the gain does not arise merely from trading distortion for stronger generative synthesis. Instead, the residual-informed adaptation allows the pretrained flow prior to produce details that remain more consistent with the decoded video content.

C.2 PSNR Results at 1080p

Figure 9 further reports the PSNR–bitrate results at 1080p. RIFT consistently improves upon GLC-Video over their common bitrate ranges on HEVC-B, UVG, and MCL-JCV, showing that the proposed restoration does not obtain its perceptual advantage at the cost of additional pixel-level distortion relative to this generative baseline.

The PSNR improvement is nevertheless more moderate than the gains observed in perceptual metrics and qualitative comparisons. This is consistent with the design of RIFT: its lightweight interface modulates the direction and magnitude of the update predicted by the pretrained Video DiT, rather than introducing a large auxiliary reconstruction module dedicated to pixel-wise refinement. The model therefore prioritizes perceptually plausible and content-consistent detail restoration, while retaining the decoded latent as the structural anchor. Consequently, the main advantage of RIFT is more pronounced in perceptual quality than in PSNR, which primarily measures pointwise reconstruction error.

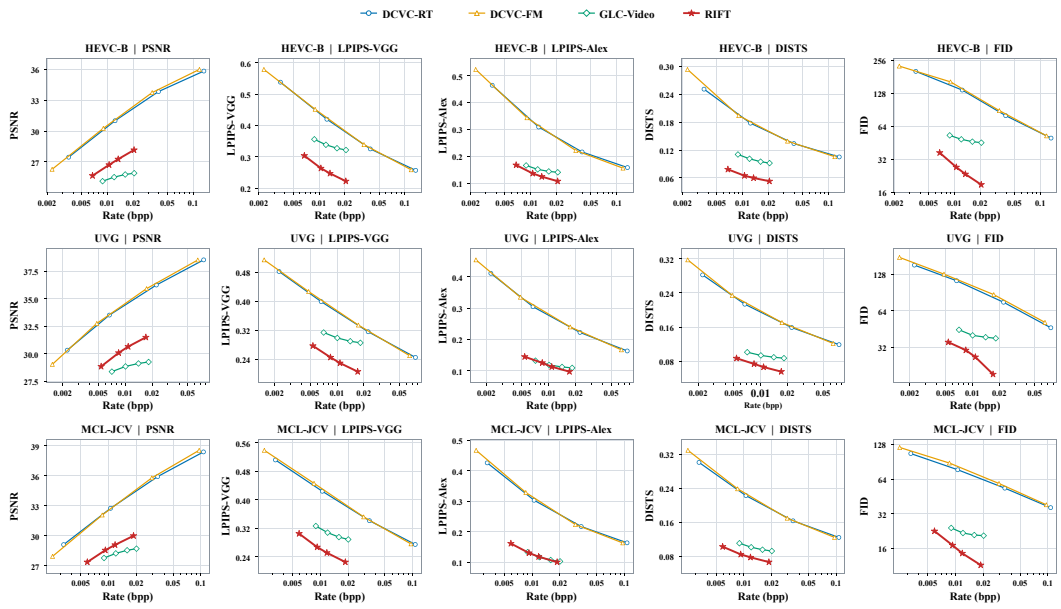

Figure 8: Rate–quality comparisons at 720p on HEVC-B, UVG, and MCL-JCV. DCVC-RT, DCVC-FM, and GLC-Video are locally reproduced and evaluated under the same test setting. The reported metrics are PSNR, LPIPS-VGG, LPIPS-Alex, DISTS, and FID. Higher PSNR and lower values of the perceptual metrics indicate better performance.

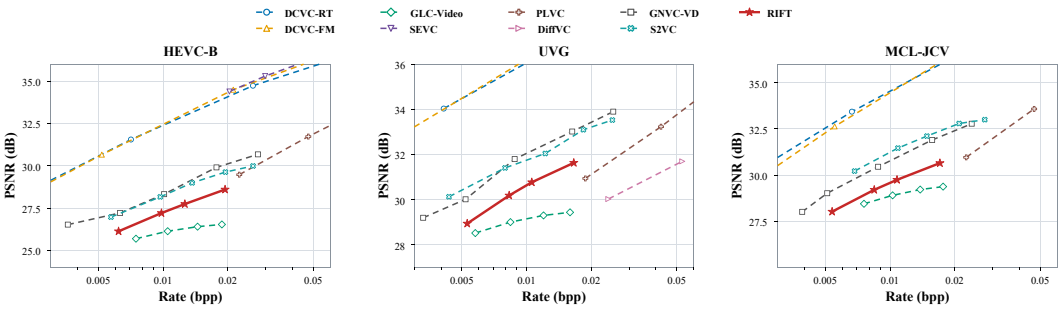

Figure 9: PSNR–bitrate comparisons at 1080p on HEVC-B, UVG, and MCL-JCV. RIFT improves upon GLC-Video over their overlapping bitrate ranges, although its gains are more pronounced in perceptual quality than in pixel-level fidelity. Higher PSNR is better.

D Qualitative Comparison with Codec-only Reconstruction

We compare RIFT with codec-only reconstruction to illustrate the visual effect of generative restoration. Both use the same codec and identical bitstreams. Codec-only directly decodes the compressed latent, whereas RIFT decodes the restored latent. The codec-only baseline therefore bypasses the DiT and all restoration heads. No model is retrained for this comparison.

Figure 10 presents five UVG sequences at 1080p: Beauty, HoneyBee, Bosphorus, ReadySteadyGo, and ShakeNDry. We use the lowest measured rate setting (QP 63) and show frame 48 (zero-based) from each sequence. The first 96 frames are coded with GOP lengths of 25, 25, 25, and 21. Within each row, all methods show the same frame and spatial crop. The enlarged regions illustrate how generative restoration improves local structures and fine textures over direct codec-only reconstruction in these examples, at identical bitrate.

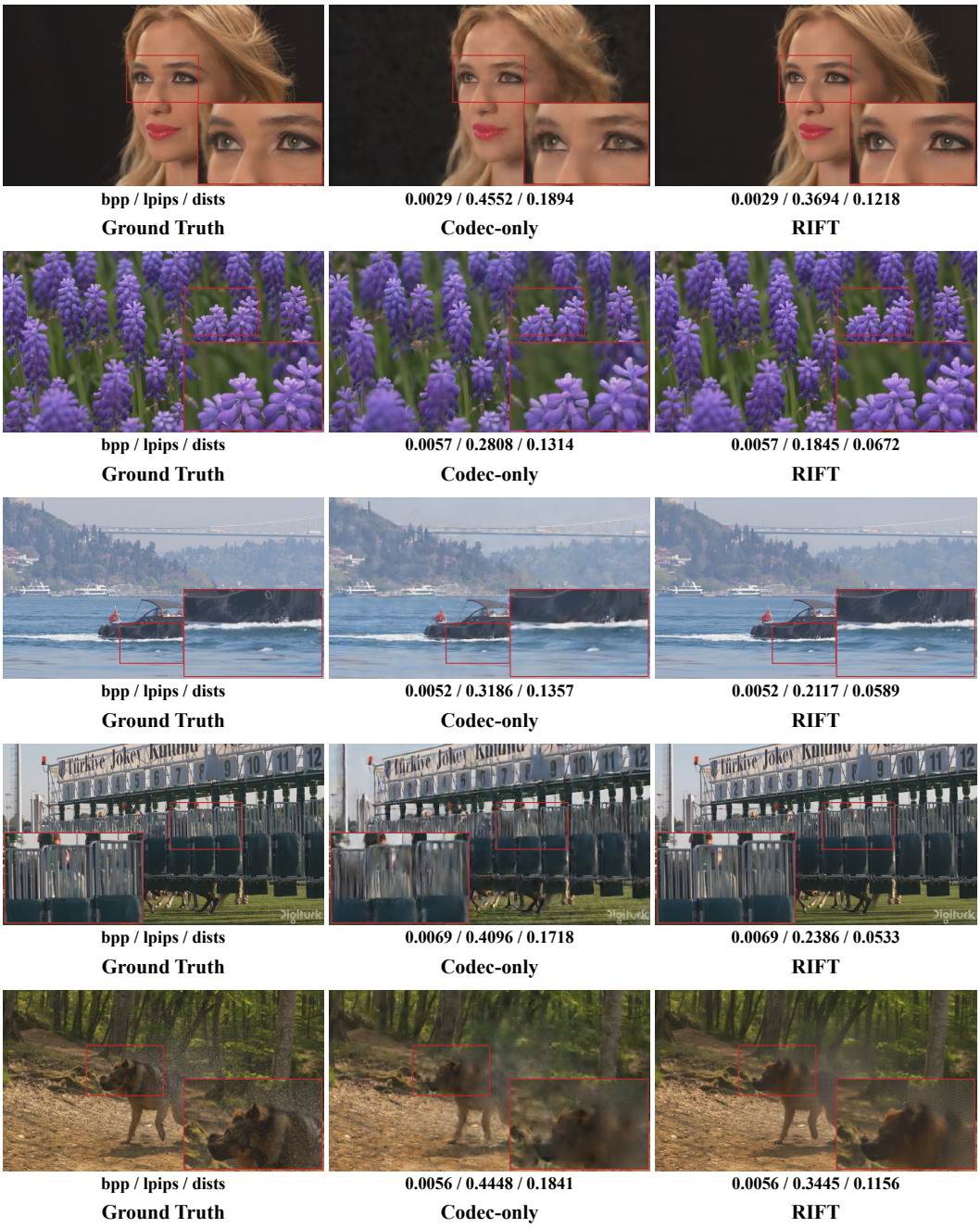

Figure 10: Qualitative comparison with codec-only reconstruction at QP 63. Rows show Beauty, HoneyBee, Bosphorus, ReadySteadyGo, and ShakeNDry; columns show ground truth, codec-only reconstruction, and RIFT. Red boxes mark the regions enlarged in the insets. Codec-only and RIFT use identical bitstreams. Labels report actual sequence bpp, full-frame LPIPS-VGG, and full-frame DISTS, respectively. Bpp uses the coded 1088×1920 canvas; quality metrics are computed on floating-point reconstructions cropped to 1080×1920. Displayed images are exported as lossless frames.

E LIMITATIONS AND FUTURE WORK

RIFT demonstrates that codec-residual restoration can be effectively guided by explicitly modulating the direction and magnitude of the generative update. However, the current model primarily validates this idea rather than defining its performance limit. The modulation network contains only $1.69\,\mathrm{M}$ parameters, a small fraction of the Video DiT backbone. More advanced architectures and training strategies may allow the DiT itself to learn this residual-aware adaptation implicitly.

Several limitations remain. First, the current mapping from operating points to flow timesteps is fixed and may not be optimal for different video content or degradation patterns. Second, each GOP is processed independently, without transferring temporal information across GOP boundaries, which may limit long-term temporal consistency. Future work will therefore explore adaptive timestep selection, implicit residual-aware adaptation within the generative backbone, and cross-GOP temporal modeling. We also plan to distill and optimize the VAE to reduce its computational cost and further improve end-to-end reconstruction speed.