Spatiotemporal Diffusion Priors for Extreme Video Compression

Lucas Relic[1,2], André Emmenegger[1], Roberto Azevedo[2], Yang Zhang[2], Markus Gross[1,2], Christopher Schroers[2]

[1]ETH Zürich, Zürich, Switzerland [2]DisneyResearch|Studios, Zürich, Switzerland

***Abstract*—Diffusion models have recently demonstrated impressive results in image compression, where the strong spatial prior enables the synthesis of fine details rather than allocating bits to transmit them. In this work, we propose to extend this paradigm to video compression by utilizing a generative spatiotemporal prior and present the first codec based on a video diffusion model. Our method operates by performing long-context interpolation guided by sparse inter-frame predictions, thus requiring minimal motion information. To this end, we develop a sparse, bidirectional optical flow which serves as a bitrate-efficient motion conditioning in the diffusion decoding process. The resulting codec can compress videos to extremely low rates (as low as 0.01 bits per pixel) while maintaining realistic textures and motion, and outperforms both neural and traditional baselines on several benchmark datasets. Our method shows state-of-the-art performance in perceptually-oriented distortion metrics, and, when considering rate-realism, we achieve an improvement in FID score of up to 73.3 at the same bitrate compared to the leading traditional video codec, VTM. Overall, we present an important first work examining spatiotemporal diffusion priors for video compression.**

***Index Terms*—video compression, generative modeling, diffusion models.**

I. Introduction

Traditional image and video codecs, particularly at extremely low bitrates, typically suffer from pronounced visual artifacts such as blurring, blocking, and banding (Fig. 1, center and right). To address these issues, recent research in the image domain has shifted its focus toward prioritizing perceptual quality and realism [1], [2]. This is achieved by leveraging powerful generative spatial priors, such as diffusion models, to synthesize high-frequency details and textures which are otherwise expensive to transmit [1], [2].

Recent neural video compression (NVC) methods have adopted a similar approach to utilize generative priors [3]–[5]. However, notably, those methods only apply such a prior *spatially* within a single frame. Indeed, existing video compression methods, both traditional [6], [7] and learned [3], [5], [8], [9] ones, process content on a frame-by-frame basis and extract motion vectors from the previous few frames as context when encoding the current frame [8]–[10]. H.265 [6] and VTM [7] use this context to generate frame predictions for residual coding, while NVC methods, such as the DCVC family [8]–[10], use it to update probability tables used for entropy coding. DiffVC [5] adopts the same paradigm, but uses a diffusion decoder to produce finer details. I^{2}VC [4] employs a unique approach, implementing motion compensation by performing a masked DDIM inversion strategy on previous features. However, due to their sequential operation, all these methods can suffer from temporal inconsistencies between frames and errors in motion estimation can propagate to the output reconstruction, affecting final output quality. Furthermore, these approaches critically lack the ability to incorporate a global prior over motion dynamics.

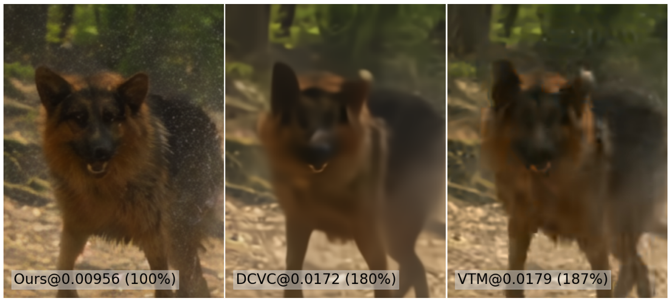

Fig. 1. Reconstructions from our method (left) contain significantly more detail and appear more realistic than those from state-of-the-art neural (center, DCVC-FM [9]) or traditional (right, VTM [7]) video compression codecs while using substantially fewer bits. Frames are annotated with "Method@BPP", where BPP is the rate in which the video was encoded in bits per pixel (rate is also shown as a percentage of our method).

Unlike previous work, in this paper, we propose using a video diffusion model as a spatial *and temporal* prior, capable of synthesizing plausible motion and thus reducing the need to transmit such information. Video diffusion models [11] emerge as a top choice in this context, providing a strong prior that has been proven effective for a variety of tasks, such as motion retargeting [12]–[14] or frame interpolation [11]. To the best of our knowledge, our method is the first to explore a video diffusion model for video compression.

Our novel video codec leverages a video diffusion model to reconstruct the source video at the receiver by operating on an entire group of pictures (GOP) at once, ensuring temporal consistency and allowing it to infer motion dynamics from the entire sequence, rather than a small frame buffer. As a result, our method operates with minimal motion information, which we achieve by developing a sparse, bidirectional optical flow to guide the reconstruction process. Using only two keyframes and this sparse flow, our codec can compress videos to extremely low bitrates (as low as 0.01 bits per pixel). The proposed approach achieves state-of-the-art performance in both rate-realism and rate-distortion (as measured by perceptually oriented metrics), demonstrating the potential of generative spatiotemporal priors for next-generation video compression.

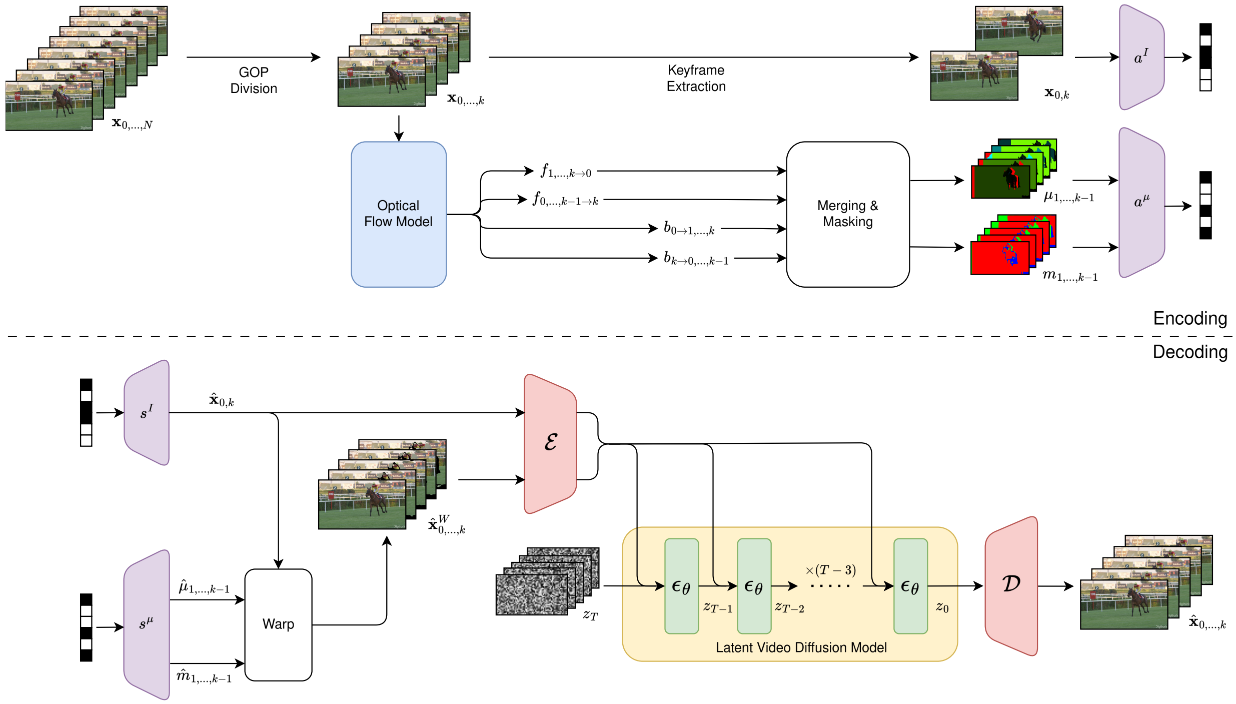

Fig. 2. Architecture and coding process of our proposed method. μ is our bidirectional optical flow map, m is the corresponding flow mask, $f_{i\to j}$ is optical flow from frame i to j, and $\hat{\mathbf{x}}^W$ is our warped inter-frame prediction. Variables denoted with $\hat{}$ (*e.g.* $\hat{\mu}$) are compressed reconstructions. a and s are analysis and synthesis transforms, respectively, of a compressive autoencoder, and $\mathcal{E}$ and $\mathcal{D}$ are the encoder and decoder, respectively, of a latent video diffusion model.

II. Method

At a high level, our proposed codec employs a conditional video diffusion model to perform long-context interpolation between keyframes, further guided by spatially aligned predictions of intermediate frames produced from sparse, bidirectional optical flow maps. The sparse flow can be efficiently compressed to low rates and ensures accurate object positions from the source video, while the diffusion model leverages its spatiotemporal prior to synthesize a detailed video with realistic motion and natural textures.

A. Encoding and Decoding

Fig. 2 shows the encoding and decoding process of our method.

At the encoder, the video is first split into GOPs of k frames each. The first and last frames of each GOP are selected as keyframes and transmitted to the receiver via a diffusion-based image codec [2]. Simultaneously, all frames of the GOP are processed with an optical flow model [15], which produces flow maps from each intermediate frame to the first and last keyframes. Both flow maps are then passed to a flow merging and masking module, which produces our proposed bidirectional optical flow maps and corresponding trinary masks for each predicted frame (see Sec. II-B). The flow maps and masks are then encoded with a temporally-aware hyperprior compression network, and transmitted to the receiver.

At the decoder, the compressed keyframes are backwards warped according to the bidirectional flow and masks to produce inter-frame predictions $\hat{\mathbf{x}}^W$ (visualized in Fig. 3). In areas with no flow information, $\hat{\mathbf{x}}^W$ is set to zero. The keyframes and inter-frame predictions are then encoded to a latent space and given as input to the denoising network (see Sec. II-C). Finally, the video latents are reconstructed by the diffusion model and decoded back to the image space to produce the final frame reconstructions.

B. Bidirectional Optical Flow

Our flow extraction process aims to compute an efficient representation of inter-frame motion to act as additional guidance to the diffusion process. Dense optical flow has been proven to be effective as motion control for video generation tasks [12], [13]; however, this dense flow is too costly to compress, especially when targeting very low bitrates. We propose to address this issue by: *1)* constructing a merged, bidirectional optical flow for each predicted frame, combining motion information from multiple keyframes in a unified representation, and *2)* sparsifying this bidirectional flow in areas where it is not accurate, allowing for better compression rates without introducing errors.

To build our bidirectional flow, we start by computing both forward and backward flow between each intermediate frame and both keyframes of the GOP:

Forward flow: $f_{i\to\{0,k\}}$ from $i \to 0$ or k, respectively

Backward flow: $b_{\{0,k\}\to i}$ from 0 or $k \to i$, respectively

Only the forward flows are transmitted and used in the decoding process. The backward flows, available only at the encoder side, are used to perform a forward-backwards consistency check [16] to validate the computed flows. A flow vector at pixel $p = (x, y)$ in $f_{i\to 0}$ is marked valid if:

$$\|f_{i\to 0}(p) \ + \ b_{0\to i}(p + f_{i\to 0}(p))\|_2 < \tau \tag{1}$$

for $i \in \{1, ..., k-1\}$, where τ is a predefined threshold. We similarly perform this consistency check for $f_{i\to k}$. This check yields the most accurate flows to either keyframe, which we then combine into a unified representation by building a single *bidirectional* flow map (Fig. 3, μ).

During this validity check, we additionally mask (i.e., sparsify) the bidirectional flow in areas where neither $f_{i\to 0}$ nor $f_{i\to k}$ is valid. The main benefit of this masking is that inaccurate flow values would propagate error to the final reconstruction. In contrast, the holes introduced by masking are easily filled by the diffusion model (Fig. 3, $\hat{\mathbf{x}}^W$ vs. $\hat{\mathbf{x}}$). Formally, we construct a combined flow field μ_i and trinary mask m_i where each pixel location p is defined as:

$$\mu_i(p) = \begin{cases} f_{i\to 0}(p), & \text{if valid} \\ f_{i\to k}(p), & \text{if } f_{i\to 0} \text{ invalid, but } f_{i\to k} \text{ valid} \\ \emptyset, & \text{otherwise} \end{cases} \tag{2}$$

$$m_i(p) = \begin{cases} 0, & \text{if } \mu_i(p) \text{ from } f_{i\to 0} \\ 1, & \text{if } \mu_i(p) \text{ from } f_{i\to k} \\ 2, & \text{if invalid} \end{cases} \tag{3}$$

C. Conditional Video Diffusion Generation

To reconstruct the encoded video at the receiver side, we design a video diffusion model that performs long-context interpolation, constrained to the correct inter-frame motion via additional motion conditioning. We inject both the keyframes (the first and last frames of each GOP) and our inter-frame prediction $\hat{\mathbf{x}}^W$ to the diffusion model by concatenating them into the noisy diffusion input at every denoising step. The reconstructed video is then generated by the diffusion model, with faithful textures propagated from the keyframes, accurate motion inferred from $\hat{\mathbf{x}}^W$, and a realistic appearance ensured by the spatiotemporal diffusion prior.

For computational efficiency, we use a latent video diffusion model, which performs the denoising operation in the latent space of a VAE. Thus, we encode both keyframes and the inter-frame predictions to a latent space before conditioning the diffusion model. Formally, each denoising iteration is defined as:

$$z_{t-1} = \epsilon_\theta(\text{concat}(z_t, \mathcal{E}(\hat{\mathbf{x}}_0), \mathcal{E}(\hat{\mathbf{x}}_k), \mathcal{E}(\hat{\mathbf{x}}^W)), t) \tag{4}$$

where concat$(\cdot)$ is concatenation along the channel dimension.

D. Implementation

a) Architecture: To leverage a powerful generative prior while maintaining a reasonable training compute budget, we implement our video diffusion model based on Stable Video Diffusion (SVD) [11], an image-to-video diffusion model that generates videos with a resolution of 1024×576 pixels. We modify the denoising network architecture to accept our provided keyframe and inter-frame prediction conditioning signals. Following related work [17], we increase the channel dimension of the first network layer to match our new input size, and initialize the additional layers by duplicating and scaling the weights of the first layer (to mitigate excessive activation magnitudes [17]).

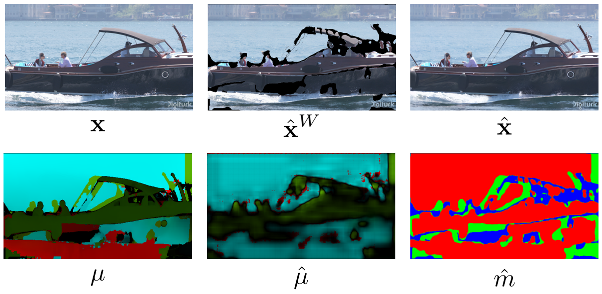

Fig. 3. Visualization of the source frame $\mathbf{x}$, our bidirectional flow map (uncompressed μ and compressed $\hat{\mu}$), mask $\hat{m}$, inter-frame prediction $\hat{\mathbf{x}}^W$, and reconstructed frame $\hat{\mathbf{x}}$. Invalid optical flow regions are masked out ($\hat{m}$, blue regions) and realistically inpainted during the diffusion process when reconstructing the output frame. Errors in $\hat{\mathbf{x}}^W$ are also corrected during the diffusion decoding process.

To compress the keyframes from each GOP, we use a pretrained diffusion-based image codec [2]. This network allows for compressing to low rates while maintaining a high level of detail, which is propagated to the predicted frames during our diffusion decoding process. Our bidirectional flow maps and masks are compressed using distinct hyperprior compression networks [18]. We extend these networks to be temporally aware by additionally passing the previous reconstructed frame, latent, and hyper-latent to their corresponding modules, accounting for temporal redundancy between frames.

b) Training: We train the flow and mask compression and video diffusion models of our method independently. We optimize all models on the **YouHQ** dataset [19] and follow Blattman *et al.* [11] to include only high-quality samples by removing static sequences and scene cuts.

The mask and flow compression networks are optimized with a standard rate-distortion loss. Pixelwise MSE is used to measure the distortion of the flow maps. Due to the discrete nature of the masks, we use a cross-entropy loss as the distortion metric for the mask compression network.

The training of the diffusion model is divided into three stages. In the first stage, we pretrain at low resolution (576×320) with *uncompressed* flows and masks, which is an easier task for the model to adapt to. The second stage is at full resolution (1024×576), still with uncompressed flows and masks. For the last training stage, we fine-tune the model at full resolution with *compressed* flows and masks, which more closely matches the inference-time task of the diffusion model. To prevent catastrophic forgetting during diffusion model training, we fully optimize only the first input layer of the denoising network and apply rank 32 LoRA fine-tuning [20] to the feedforward layers and query, key, value, and output projection layers in the attention layers. We train with a batch size of 4, a learning rate of $1 \cdot 10^{-5}$ for the LoRA parameters, and use a GOP size of 14 frames.

TABLE I

Quantitative rate-distortion (LPIPS and DISTS) and rate-realism (FID and KID) evaluation, measured by BD-quality with VTM (LD) as the anchor. Best performing is marked in bold and second underlined. Lower is better.

	BD-LPIPS (↓)			BD-DISTS (↓)			BD-FID (↓)			BD-KID (↓)		
	UVG	MCL	HEVC B	UVG	MCL	HEVC B	UVG	MCL	HEVC B	UVG	MCL	HEVC B
H.265	0.0834	0.0656	0.0684	0.0234	0.0129	0.0120	32.8	28.8	19.1	0.0167	0.0193	0.0144
VTM (LD)	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
VTM (RA)	-0.0351	-0.0399	-0.0468	-0.0214	-0.0223	-0.0235	-13.5	-12.6	-16.9	-0.00552	-0.00675	-0.00913
DCVC-FM	0.0198	0.0151	0.0189	0.0296	0.0233	0.0196	25.7	12.8	21.4	0.00846	0.00446	0.0119
Ours	**-0.0634**	**-0.0720**	**-0.100**	**-0.0757**	**-0.0878**	**-0.0878**	**-39.4**	**-47.8**	**-73.3**	**-0.0148**	**-0.0213**	**-0.0386**

III. Experiments

A. Test sequences, baselines, and metrics

We conduct experiments on multiple widely used benchmark datasets, specifically **UVG** [21], **HEVC Class B** [22], and **MCL JCV** [23]. Notably, however, the large memory requirement of video diffusion models causes evaluation on high-resolution datasets to be infeasible, and we therefore downscale all evaluation datasets to 1024×576 pixels.

Our method is evaluated against the state-of-the-art neural baseline **DCVC-FM** [9] and leading traditional codecs **VTM** [7] and **H.265** [6]. We use two configurations of VTM: *random access* (RA) and *low delay P* (LD), and take x265 *veryslow* implementation of H.265. Notably, we are limited to only open-source baselines (due to the nonstandard resolution of our evaluation datasets), and therefore, we are unable to compare to generative baselines, as the source code for these methods is unavailable. As we focus on video compression to low rates, all evaluated codecs use a target bitrate between 0.01 and 0.03 bpp.

Following related work on generative compression [2], [3], [5], we evaluate on **LPIPS** and **DISTS** and measure realism with **FID** and **KID**. Quantitative evaluation of generative compression methods is known to be a challenging problem [2], [24], and it has been proven that achieving low quantitative distortion (*i.e.*, metrics like PSNR) *necessarily* reduces the performance in metrics measuring perceptual quality or realism [24]. We therefore prioritize perceptual quality and use metrics that more closely model human perception rather than pixelwise distortion metrics such as PSNR. We also provide visual examples in Fig. 4 for qualitative analysis.

B. Results

Quantitative rate-distortion and rate-realism results are shown in Table I, measured by Bjontgaard-Delta quality [25] (BD-quality) with VTM (LD) as the anchor. Across all experimental conditions, our method consistently outperforms competing approaches, yielding an average improvement of 0.100 in LPIPS score on the HEVC B dataset compared to VTM (LD) – more than twice the gain achieved by the leading video codec baseline, VTM (RA). In particular, our method shows the most significant performance increase when considering realism, achieving FID score reductions of up to 73.3 at equivalent bitrates, an improvement of 56.4 compared to VTM (RA). The benefit of generative models is particularly clear in this setting, due to their tendency to produce realistic reconstructions on the natural video manifold.

Visual examples of our method compared to the baselines are shown in Fig. 4. The reconstructions from our method are consistently more detailed than the neural and traditional baselines, particularly with respect to textures, as can be seen in the tile in row 1 and the tree bark in row 2. VTM and H.265 also suffer from unrealistic banding or blocking artifacts, which are not present in our reconstructions. These results confirm our hypothesis that a strong generative prior can improve compression codecs at very low bitrate settings.

C. Ablation studies

To examine the impact of each component on overall performance, we ablate the inter-frame prediction conditioning signals and third-stage finetuning (using compressed flows and masks) from our method. Experiments are performed on the HEVC B dataset and evaluated with LPIPS, measured in BD-rate [25] with our full method as the anchor.

Removing the inter-frame predictions results in a BD-rate of 20.3%, indicating that this conditioning signal is critical to our model's low-rate performance. Our third finetuning stage is also necessary to achieve high performance, as ablating it resulted in a BD-rate of 38.7%, which suggests the diffusion model successfully learns to apply its inherent spatiotemporal prior to correct errors caused by compressing the provided conditioning signals (shown in Fig. 3, top row).

IV. Conclusion

We present the first video diffusion-based compression codec, which utilizes the strong spatiotemporal prior from the video diffusion model to produce detailed reconstructions with realistic motion. Utilizing such a strong prior enables our method to achieve SoTA rate-realism and rate-distortion performance, particularly at very low bitrates. While our approach demonstrates promising results in the low-rate regime, its performance is less effective at higher bitrates (above 0.03 bits per pixel), and is currently limited by the computational cost of diffusion-based inference, in terms of both runtime and hardware requirements. While improving diffusion efficiency is orthogonal to the focus of this work, it remains an important area for future research. Additional directions for future work include exploring more compact inter-frame motion representations and extending the framework to support a broader effective bitrate range.

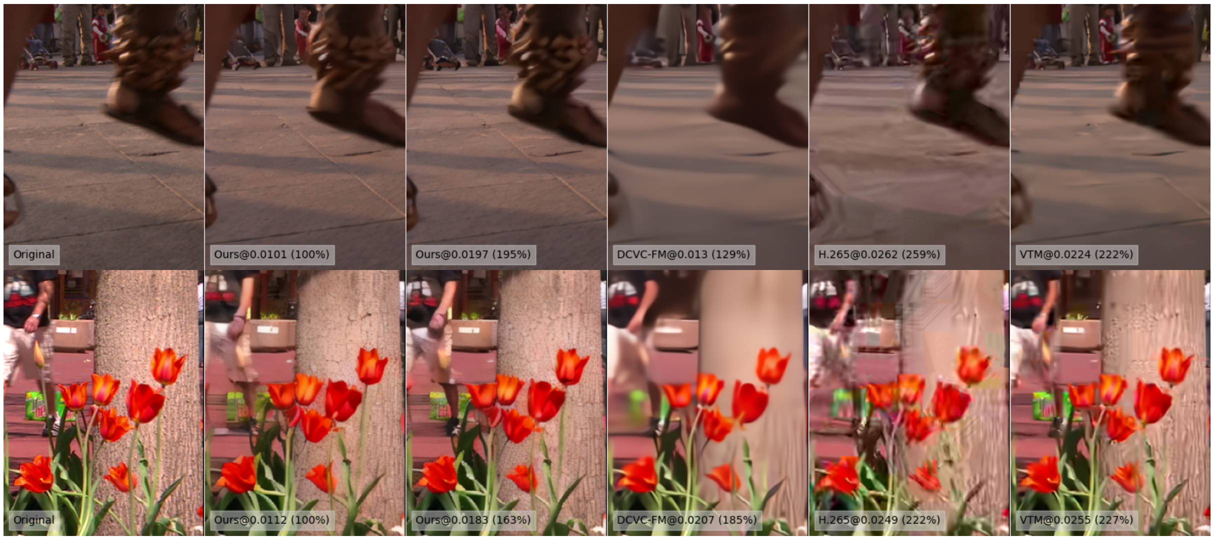

Fig. 4. Qualitative comparison of our method (2nd and 3rd columns) to a state-of-the-art neural codec (DCVC-FM [9]) and traditional video codecs (H.265 [6] and VTM *random access* [7]). Our method produces more details, avoiding unnatural artifacts and always producing realistic images. Frames are annotated with "Method@BPP", where BPP is the rate in which the video was encoded in bits per pixel (rate is also shown as a percentage of our method's lowest rate). (Best viewed digitally.)

References

[1] R. Yang and S. Mandt, "Lossy Image Compression with Conditional Diffusion Models," *Advances in Neural Information Processing Systems*, vol. 36, pp. 64 971–64 995, Dec. 2023.

[2] L. Relic, R. Azevedo, Y. Zhang, M. Gross, and C. Schroers, "Bridging the gap between gaussian diffusion models and universal quantization for image compression," in *Proceedings of the Computer Vision and Pattern Recognition Conference (CVPR)*, June 2025, pp. 2449–2458.

[3] F. Mentzer, E. Agustsson, J. Ballé, D. Minnen, N. Johnston, and G. Toderici, "Neural Video Compression Using GANs for Detail Synthesis and Propagation," in *Computer Vision – ECCV 2022*. Cham: Springer Nature Switzerland, 2022, pp. 562–578.

[4] M. Liu, C. Xu, Y. Gu, C. Yao, and Y. Zhao, "I$ˆ2$VC: A Unified Framework for Intra- & Inter-frame Video Compression," Jun. 2024.

[5] W. Ma and Z. Chen, "Diffusion-based Perceptual Neural Video Compression with Temporal Diffusion Information Reuse," Jan. 2025.

[6] "H.265/hevc," https://hevc.hhi.fraunhofer.de/.

[7] "Vtm," https://vcgit.hhi.fraunhofer.de/jvet/VVCSoftware_VTM.

[8] J. Li, B. Li, and Y. Lu, "Deep Contextual Video Compression," in *Advances in Neural Information Processing Systems*, vol. 34. Curran Associates, Inc., 2021, pp. 18 114–18 125.

[9] ——, "Neural Video Compression with Feature Modulation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 26 099–26 108.

[10] ——, "Hybrid Spatial-Temporal Entropy Modelling for Neural Video Compression," in *Proceedings of the 30th ACM International Conference on Multimedia*, ser. MM '22. New York, NY, USA: Association for Computing Machinery, Oct. 2022, pp. 1503–1511.

[11] A. Blattmann, T. Dockhorn, S. Kulal, D. Mendelevitch, M. Kilian, D. Lorenz, Y. Levi, Z. English, V. Voleti, A. Letts, V. Jampani, and R. Rombach, "Stable Video Diffusion: Scaling Latent Video Diffusion Models to Large Datasets," Nov. 2023.

[12] X. Shi, Z. Huang, F.-Y. Wang, W. Bian, D. Li, Y. Zhang, M. Zhang, K. C. Cheung, S. See, H. Qin, J. Dai, and H. Li, "Motion-I2V: Consistent and Controllable Image-to-Video Generation with Explicit Motion Modeling," in *ACM SIGGRAPH 2024 Conference Papers*, ser. SIGGRAPH '24. New York, NY, USA: Association for Computing Machinery, Jul. 2024, pp. 1–11.

[13] S. Yin, C. Wu, J. Liang, J. Shi, H. Li, G. Ming, and N. Duan, "DragNUWA: Fine-grained Control in Video Generation by Integrating Text, Image, and Trajectory," Aug. 2023.

[14] M. Kansy, J. Naruniec, C. Schroers, M. Gross, and R. M. Weber, "Reenact Anything: Semantic Video Motion Transfer Using Motion-Textual Inversion," Aug. 2024.

[15] Z. Teed and J. Deng, "RAFT: Recurrent All-Pairs Field Transforms for Optical Flow," in *Computer Vision – ECCV 2020*, A. Vedaldi, H. Bischof, T. Brox, and J.-M. Frahm, Eds. Cham: Springer International Publishing, 2020, pp. 402–419.

[16] S. Meister, J. Hur, and S. Roth, "Unflow: Unsupervised learning of optical flow with a bidirectional census loss," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 32, no. 1, 2018.

[17] B. Ke, A. Obukhov, S. Huang, N. Metzger, R. C. Daudt, and K. Schindler, "Repurposing Diffusion-Based Image Generators for Monocular Depth Estimation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 9492–9502.

[18] J. Ballé, D. Minnen, S. Singh, S. J. Hwang, and N. Johnston, "Variational image compression with a scale hyperprior," in *International Conference on Learning Representations*, Feb. 2018.

[19] S. Zhou, P. Yang, J. Wang, Y. Luo, and C. C. Loy, "Upscale-A-Video: Temporal-Consistent Diffusion Model for Real-World Video Super-Resolution," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 2535–2545.

[20] E. J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, W. Chen *et al.*, "Lora: Low-rank adaptation of large language models." *ICLR*, vol. 1, no. 2, p. 3, 2022.

[21] A. Mercat, M. Viitanen, and J. Vanne, "UVG dataset: 50/120fps 4K sequences for video codec analysis and development," in *Proceedings of the 11th ACM Multimedia Systems Conference*, ser. MMSys '20. New York, NY, USA: Association for Computing Machinery, May 2020, pp. 297–302.

[22] F. Bossen, J. Boyce, X. Li, V. Seregin, and K. Sühring, "Vtm common test conditions and software reference configurations for sdr video," 2020. [Online]. Available: https://jvet-experts.org/doc_end_user/current_document.php?id=10545

[23] "MCL-JCV: A JND-based H.264/AVC video quality assessment dataset | IEEE Conference Publication | IEEE Xplore," https://ieeexplore.ieee.org/abstract/document/7532610.

[24] Y. Blau and T. Michaeli, "Rethinking Lossy Compression: The Rate-Distortion-Perception Tradeoff," in *Proceedings of the 36th International Conference on Machine Learning*. PMLR, May 2019, pp. 675–685.

[25] K. Andersson, R. Sjöberg, and A. Norkin, "Reliability metric for bd measurement," ITU-T SG16 Q.6 Document, VCEG-AL22, 2009.