WF-VAE: Enhancing Video VAE by Wavelet-Driven Energy Flow for Latent Video Diffusion Model

Zongjian Li[1,3,*], Bin Lin[1,3,*], Yang Ye[1,3], Liuhan Chen[1,3], Xinhua Cheng[1,3], Shenghai Yuan[1,3], Li Yuan[1,2,†]

[1]Peking University, [2]Peng Cheng Laboratory, [3]Rabbitpre Intelligence

Abstract

*Video Variational Autoencoder (VAE) encodes videos into a low-dimensional latent space, becoming a key component of most Latent Video Diffusion Models (LVDMs) to reduce model training costs. However, as the resolution and duration of generated videos increase, the encoding cost of Video VAEs becomes a limiting bottleneck in training LVDMs. Moreover, the block-wise inference method adopted by most LVDMs can lead to discontinuities of latent space when processing long-duration videos. The key to addressing the computational bottleneck lies in decomposing videos into distinct components and efficiently encoding the critical information. Wavelet transform can decompose videos into multiple frequency-domain components and improve the efficiency significantly, we thus propose **Wavelet Flow VAE (WF-VAE)**, an autoencoder that leverages multi-level wavelet transform to facilitate low-frequency energy flow into latent representation. Furthermore, we introduce a method called **Causal Cache**, which maintains the integrity of latent space during block-wise inference. Compared to state-of-the-art video VAEs, WF-VAE demonstrates superior performance in both **PSNR** and **LPIPS** metrics, achieving **2×** higher throughput and **4×** lower memory consumption while maintaining competitive reconstruction quality. Our code and models are available at https://github.com/PKU-YuanGroup/WF-VAE.*

1. Introduction

The release of Sora [5], a video generation model developed by OpenAI, has pushed the boundaries of synthesizing photorealistic videos, drawing unprecedented attention to the field of video generation. Recent advancements in Latent Video Diffusion Models (LVDMs), such as Open-Sora Plan [19], Open-Sora [49], CogVideoX [40], EasyAnimate [39], Movie Gen [25], and ConsisID [44], have led to substantial improvements in video generation quality.

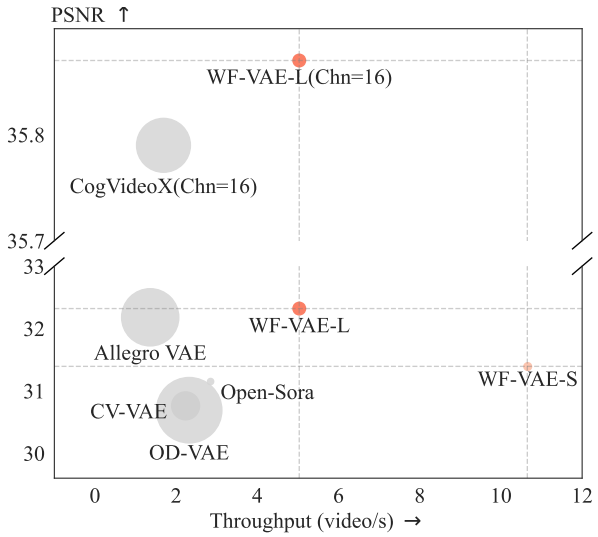


Figure 1. **Performance comparison of video VAEs.** Bubble area indicates the memory usage during inference. All measurements are conducted on 33 frames with 256×256 resolution videos. "Chn" represents the number of latent channels. Higher PSNR and throughput indicate better performance.

These methods establish a compressed latent space [27] using a pre-trained video Variational Autoencoder (VAE), where the compression quality fundamentally determines the generative performance.

Current video VAEs remain constrained by fully convolutional architectures inherited from the image era. They address video flickering and redundant information by incorporating spatio-temporal interaction and spatio-temporal compression layers. Several recent works, including OD-VAE [6, 19], CogVideoX [40], CV-VAE [48], and Allegro [50] adopt dense 3D structure to achieve high-quality video compression. While these methods demonstrate impressive reconstruction performance, they require prohibitively intensive computational resources. In contrast, alternative approaches such as Movie Gen [25] and Open-Sora [49] utilize 2+1D architecture, resulting in reduced computational requirements at the cost of lower reconstruc-

*Equal contribution
†Corresponding author.

tion quality. Previous architectures have inadequately leveraged temporal redundancy in video data, necessitating the use of redundant spatio-temporal interaction layers to improve video compression quality.

Many attempts employ block-wise inference strategies to trade computation time for memory, thus addressing computational bottlenecks in processing high-resolution, long-duration videos. EasyAnimate [39] introduced Slice VAE to encode and decode video frames in groups, but this approach leads to discontinuous output videos. Several methods, including Open-Sora [49], Open-Sora Plan [19], Allegro [50], implement tiling inference strategies but often produce spatio-temporal artifacts in overlapping regions. Although CogVideoX [40] employ caching to ensure convolution continuity, its reliance on group normalization [37] disrupts the independence of temporal feature, thus preventing lossless block-wise inference. These limitations highlight two core challenges faced by video VAEs: (**1**) *excessive computational demands due to redundant architecture*, and (**2**) *compromised latent space integrity resulting from existing tiling inference strategies, which cause artifacts and flickering in reconstructed videos.*

Wavelet transform [23, 26] decomposes videos into multiple frequency-domain components. This decomposition enables prioritization strategies for encoding crucial video components. In this work, we propose **Wavelet Flow VAE (WF-VAE)**, a novel autoencoder that utilizes multi-level wavelet transforms for extracting multi-scale pyramidal features and establishes a main energy flow pathway for these features to flow into latent representation. This pathway bypasses low-frequency video information to latent space, skipping the backbone network. Our WF-VAE enables a simplified backbone design with reduced 3D convolutions, significantly reducing computational costs. To address the potential latent space disruption, we propose *Causal Cache* mechanism. This approach leverages the properties of causal convolution. It maintains the continuity of the convolution sliding window through a caching strategy, which ensures numerical identity between block-wise inference and direct inference results. Experimental results show that WF-VAE achieves state-of-the-art reconstruction quality and computational efficiency performance. To summarize, major contributions of our work include:

- We propose **WF-VAE**, which leverages multi-level wavelet transforms to extract pyramidal features and establishes a main energy flow pathway for video information flow into a latent representation.
- We introduce a lossless block-wise inference mechanism called *Causal Cache*, which maintains identical performance as direct inference across videos of any duration.
- Extensive experimental evaluations of video reconstruction and generation demonstrate that WF-VAE achieves state-of-the-art performance.

2. Related Work

Variational Autoencoders. [18] introduced the VAE based on variational inference, establishing a novel generative network structure. Subsequent research [24, 31, 35] demonstrated that training and inference in VAE latent space could substantially reduce computational costs for diffusion models. [27] further proposed a two-stage image synthesis approach by decoupling perceptual compression from diffusion model. After that, numerous studies explored video VAEs with a focus on more efficient video compression, including Open-Sora Plan [19], CogVideoX [40], and other models [3, 6, 41, 48, 50, 51]. Current video VAE architectures primarily derive from earlier image VAE design [9, 27] and use a convolutional backbone. LiteVAE [29] employs a wavelet-based encoder inspired by latent-pixel similarity but overlooks energy perspectives, causing encoding-decoding energy asymmetry.

Latent Video Diffusion Models. In the early stages of Latent Video Diffusion Models (LVDMs) development, models like AnimateDiff [13], and MagicTime [45, 46] primarily utilized the U-Net backbone [28] for denoising, without temporal compression in VAEs. Following the paradigm introduced by Sora, recent open-source models such as Open-Sora [49], Open-Sora Plan [19, 20], and CogVideoX [40] have adopted DiT backbone with a spatiotemporally compressed VAE. Some methods, including Open-Sora [49] and Movie Gen [25], employ a 2+1D design in either the DiT backbone or the VAE to reduce training costs. For LVDMs, the upper limit of video generation quality is primarily determined by the VAE's reconstruction quality. As overhead increases rapidly with scale, optimizing the VAE becomes essential for processing large-scale data and enabling extensive pre-training.

3. Method

3.1. Wavelet Transform

Preliminary. The Haar wavelet transform [23, 26], a fundamental form of wavelet transform, is widely used in signal processing [11, 12, 15, 42, 43]. It efficiently captures spatio-temporal information by decomposing signals through two complementary filters. The first is the Haar scaling filter $\mathbf{h} = \frac{1}{\sqrt{2}}[1, 1]$, which acts as a low-pass filter that captures the average or approximation coefficients. The second is the Haar wavelet filter $\mathbf{g} = \frac{1}{\sqrt{2}}[1, -1]$, which functions as a high-pass filter that extracts the detail coefficients. These orthogonal filters are designed to be simple yet effective, with the scaling filter smoothing the signal and the wavelet filter detecting local changes or discontinuities.

Multi-level Wavelet Transform. For a video signal $\mathbf{V} \in \mathbb{R}^{c \times t \times h \times w}$, where c, t, h, and w denote the number of channels, temporal frames, height, and width respectively, the 3D Haar wavelet transform at layer l is defined as:

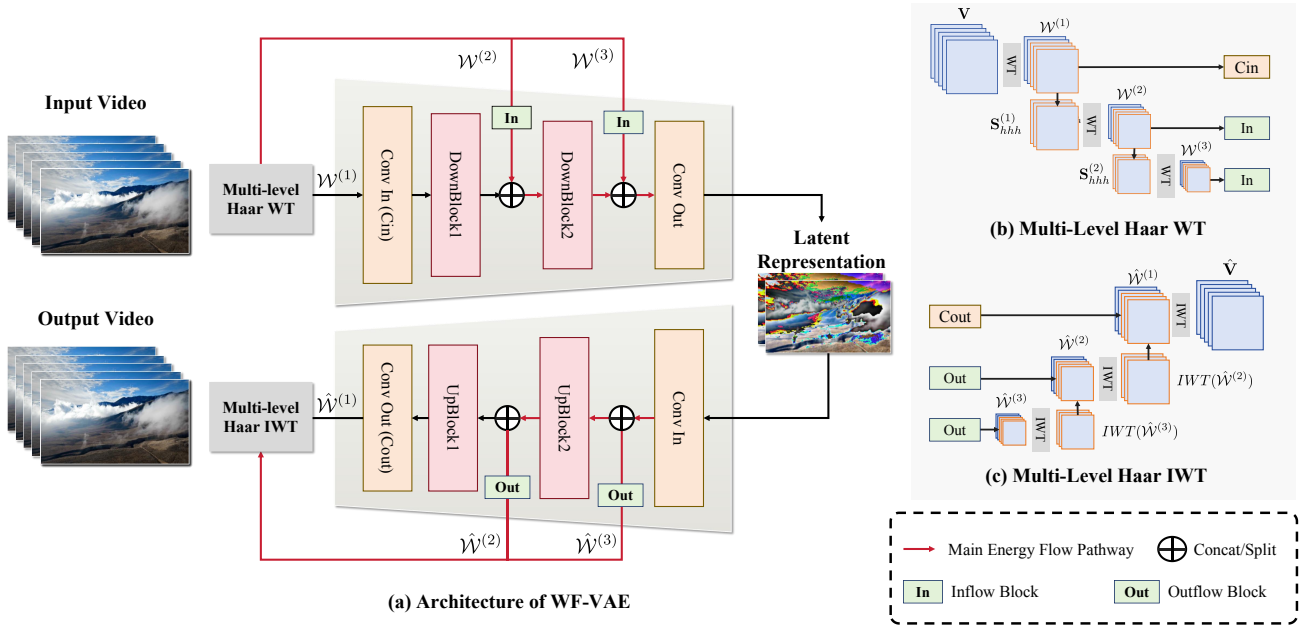


Figure 2. **Overview of WF-VAE.** Our architecture consists of a backbone and a main energy flow pathway. The pathway functions as a "highway" for the main flow of video energy, channeling this energy into the backbone through concatenations, allowing more critical video information to be preserved in the latent representation.

$$\mathbf{S}^{(l)}_{ijk} = \mathbf{S}^{(l-1)} * (f_i \otimes f_j \otimes f_k), \tag{1}$$

where $f_i, f_j, f_k \in \{\mathbf{h}, \mathbf{g}\}$ represent the filters applied along each dimension, and $*$ represents the convolution operation. The transform begins with $\mathbf{S}^{(0)} = \mathbf{V}$, and for subsequent layers, $\mathbf{S}^{(l)} = \mathbf{S}^{(l-1)}_{hhh}$, indicating that each layer operates on the low-frequency component from the previous layer. At each decomposition layer l, the transform produces eight sub-band components: $\mathcal{W}^{(l)} = \{\mathbf{S}^{(l)}_{hhh}, \mathbf{S}^{(l)}_{hhg}, \mathbf{S}^{(l)}_{hgh}, \mathbf{S}^{(l)}_{ghh}, \mathbf{S}^{(l)}_{hgg}, \mathbf{S}^{(l)}_{ggh}, \mathbf{S}^{(l)}_{ghg}, \mathbf{S}^{(l)}_{ggg}\}$. Here, $\mathbf{S}^{(l)}_{hhh}$ represents the low-frequency component across all dimensions, while $\mathbf{S}^{(l)}_{ggg}$ captures high-frequency details. To implement different downsampling rates in the temporal and spatial dimensions, a combination of 2D and 3D wavelet transforms can be implemented. Specifically, to obtain a compression rate of 4×8×8 (temporal×height×width), we can employ a combination of a two-layer 3D wavelet transform followed by a one-layer 2D wavelet transform.

3.2. Architecture Design of WF-VAE

Through analyzing different sub-bands, we find that video energy is mainly concentrated in the low-frequency sub-band $\mathbf{S}^{(1)}_{hhh}$. Based on this observation, we establish an energy flow pathway, as illustrated in Fig. 2, so that low-frequency information can smoothly flow from video to latent representation during the encoding process and then flow back to the video during the decoding process. This would inherently allow the model to pay more attention to low-frequency information and apply higher compression rates to high-frequency details. With the additional path, we can reduce the computational cost brought by the dense 3D convolutions in the backbone.

Specifically, given a video $\mathbf{V}$, we apply multi-level wavelet transform to obtain pyramid features $\mathcal{W}^{(1)}, \mathcal{W}^{(2)}$ and $\mathcal{W}^{(3)}$. We utilize $\mathcal{W}^{(1)}$ as the input for the encoder and the target output for the decoder. The backbone and multi-level wavelet transform downsample the feature maps simultaneously at every downsampling layer, enabling the concatenation of features from two branches in the backbone. We employ Inflow Block to transform the channel numbers of $\mathcal{W}^{(2)}$ and $\mathcal{W}^{(3)}$ to C_{flow}, which are then concatenated with feature maps from backbone. We compare C_{flow} to the width of the energy flow pathway, as analyzed in Sec. 4.3. On the decoder side, we maintain a structure symmetrical to the encoder. We split feature maps with C_{flow} channels from the backbone and process them through Outflow Block to obtain $\hat{\mathcal{W}}^{(2)}, \hat{\mathcal{W}}^{(3)}$. To allow information to flow from the lower layer to the hhh sub-band of the next layer, we have:

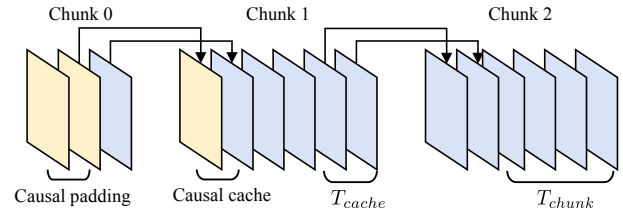
$$\hat{\mathbf{S}}^{(2)}_{hhh} = IWT(\hat{\mathcal{W}}^{(3)}) + \hat{\mathbf{S}}^{(2)}_{outflow,hhh}. \tag{2}$$

Similarly, at the decoder output layer:

$$\hat{\mathbf{S}}^{(1)}_{hhh} = IWT(\hat{\mathcal{W}}^{(2)}) + \hat{\mathbf{S}}^{(1)}_{outflow,hhh}. \tag{3}$$

Overall, we strategically design shortcuts to prioritize low-frequency information, thereby enhancing its representation within the latent space.

3.3. Causal Cache



(a) Illustration of *Casual Cache*.

(b) Qualitative comparison of tiling inference and *Causal Cache*

Figure 3. **(a)** *Causal Cache* with a temporal kernel size of 3 and stride 1. **(b)** Comparison of tiling inference and *Causal Cache*, highlighting how tiling causes locally **color and shape distortions** at overlaps, leading to global flickering in reconstructed videos.

We replace regular 3D convolutions with causal 3D convolutions [41] in WF-VAE. The causal convolution applies $k_t - 1$ temporal padding at the start with kernel size k_t. This padding strategy ensures that the first frame remains independent from subsequent frames, thus enabling the processing of images and videos within a unified architecture. Furthermore, we leverage the causal convolution properties to achieve lossless inference. We first extract the initial frame from a video with T frames. The remaining $T-1$ frames are then partitioned into temporal chunks, where T_{chunk} represents the chunk size. Let s_t denote the temporal convolutional stride, and $m = 0, 1, 2, \cdots$ represents the chunk block index. To maintain the continuity of convolution sliding windows, each chunk caches its tail frames for the next chunk. The number of cached frames is given by:

$$T_{cache}(m) = k_t + mT_{chunk} - s_t\lfloor\frac{mT_{chunk}}{s_t} + 1\rfloor. \quad (4)$$

For example, $k_t = 3, s_t = 1, T_{chunk} = 4$, the equation yields $T_{cache}(m) = 2$, as illustrated in Fig. 3a. Similarly, with $k_t = 3, s_t = 2, T_{chunk} = 4$, we obtain $T_{cache}(m) = 1$, indicating only the last frame requires to be cached. Special cases exist, such as when $k_t = 4, s_t = 3, T_{chunk} = 4$, which results in $T_{cache}(m) = (m \bmod 3 + 1)$. Fig. 3b provides a qualitative comparison between *Causal Cache* and the tiling strategy, illustrating how it effectively mitigates significant distortions in both color and shape.

3.4. Training Objective

Following the training strategies of [10, 27], our loss function combines multiple components, including reconstruction loss (comprising L1 and perceptual loss [47]), adversarial loss, and KL regularization [18]. Our model is characterized by a low-frequency energy flow and symmetry between the encoder and decoder. To maintain this architectural principle, we introduce a regularization term denoted as $\mathcal{L}_{WL}$ (WL loss), which enforces structural consistency by penalizing deviations from the intended energy flow:

$$\mathcal{L}_{WL} = |\hat{\mathcal{W}}^{(2)} - \mathcal{W}^{(2)}| + |\hat{\mathcal{W}}^{(3)} - \mathcal{W}^{(3)}|. \quad (5)$$

The final loss function is formulated as follows:

$$\mathcal{L} = \mathcal{L}_{recon} + \lambda_{adv}\mathcal{L}_{adv} + \lambda_{KL}\mathcal{L}_{KL} + \lambda_{WL}\mathcal{L}_{WL}. \quad (6)$$

We examine the impact of the weighting factor λ_{wl} in Sec. 4.3. Following [10], we implement dynamic adversarial loss weighting to balance the relative gradient magnitudes between adversarial and reconstruction losses:

$$\lambda_{\text{adv}} = \frac{1}{2}\left(\frac{\|\nabla_{G_L}[\mathcal{L}_{\text{recon}}]\|}{\|\nabla_{G_L}[\mathcal{L}_{\text{adv}}]\| + \delta}\right), \quad (7)$$

where $\nabla_{G_L}[\cdot]$ denotes the gradient with respect to last layer of decoder, and $\delta = 10^{-6}$ is used for numerical stability.

4. Experiments

4.1. Experimental Setup

Baseline Models. To assess the effectiveness of WF-VAE, we perform a comprehensive evaluation, comparing its performance and efficiency against several state-of-the-art VAE models. The models considered are: (1) OD-VAE [6], a 3D causal convolutional VAE used in Open-Sora Plan 1.2 [19]; (2) Open-Sora VAE [49]; (3) CV-VAE [48]; (4) CogVideoX VAE [40]; (5) Allegro VAE [50]; (6) SVD-VAE [4], which does not compress temporally and (7) SD-VAE [27], a widely used image VAE. Among these, CogVideoX VAE adopts a latent dimension of 16, whereas others utilize a latent dimension of 4. Notably, most VAEs have been validated on LVDMs, making them highly representative for comparison.

Dataset & Evaluation. We utilize the Kinetics-400 dataset [16] for both training and validation. For testing, we employ the Panda70M [7] and WebVid-10M [2] datasets. To comprehensively evaluate the model's reconstruction performance, we select Peak Signal-to-Noise Ratio (PSNR) [14], Learned Perceptual Image Patch Similarity (LPIPS) [47], and Structural Similarity Index Measure (SSIM) [36] as primary evaluation metrics. Additionally, we use reconstruction Fréchet Video Distance (rFVD) [34] to assess visual quality and temporal coherence. To assess our model's performance in generating results with the diffusion model, we utilize the UCF-101 [33] and SkyTimelapse [38] datasets for conditional and unconditional training 100,000 steps. Following [22, 32], we extract 16-frame

Method	TCPR	Chn	WebVid-10M				Panda-70M			
			PSNR (↑)	SSIM(↑)	LPIPS (↓)	rFVD (↓)	PSNR (↑)	SSIM (↑)	LPIPS(↓)	rFVD (↓)
SD-VAE [27]	$64(1\times8\times8)$	4	30.19	0.8377	0.0568	284.90	30.46	0.8896	0.0395	182.99
SVD-VAE [4]	$64(1\times8\times8)$	4	31.18	0.8689	0.0546	188.74	31.04	0.9059	0.0379	137.67
CV-VAE [48]	$256(4\times8\times8)$	4	30.76	0.8566	0.0803	369.23	30.18	0.8796	0.0672	296.28
OD-VAE [6]	$256(4\times8\times8)$	4	30.69	0.8635	0.0553	255.92	30.31	0.8935	0.0439	191.23
Open-Sora VAE [49]	$256(4\times8\times8)$	4	31.14	0.8572	0.1001	475.23	31.37	0.8973	0.0662	298.47
Allegro [50]	$256(4\times8\times8)$	4	32.18	**0.8963**	0.0524	209.68	31.70	**0.9158**	0.0421	172.72
WF-VAE-S (Ours)	$256(4\times8\times8)$	4	31.39	0.8737	0.0517	188.04	31.27	0.9025	0.0420	146.91
WF-VAE-L (Ours)	$256(4\times8\times8)$	4	**32.32**	0.8920	**0.0513**	**186.00**	**32.10**	0.9142	**0.0411**	**146.24**
CogVideoX-VAE [40]	$256(4\times8\times8)$	16	35.72	**0.9434**	0.0277	59.83	35.79	0.9527	0.0198	43.23
WF-VAE-L (Ours)	$256(4\times8\times8)$	16	**35.76**	0.9430	**0.0230**	**54.36**	**35.87**	**0.9538**	**0.0175**	**39.40**

Table 1. **Quantitative metrics of reconstruction performance.** Results demostrate that WF-VAE achieves state-of-the-art on reconstrcution performance comparing with other VAEs on WebVid-10M [2] and Panda70M [7] datasets. TCPR represents the token compression rate, and Chn indicates the number of latent channels. The highest result is highlighted in **bold**, and the second highest result is underlined.

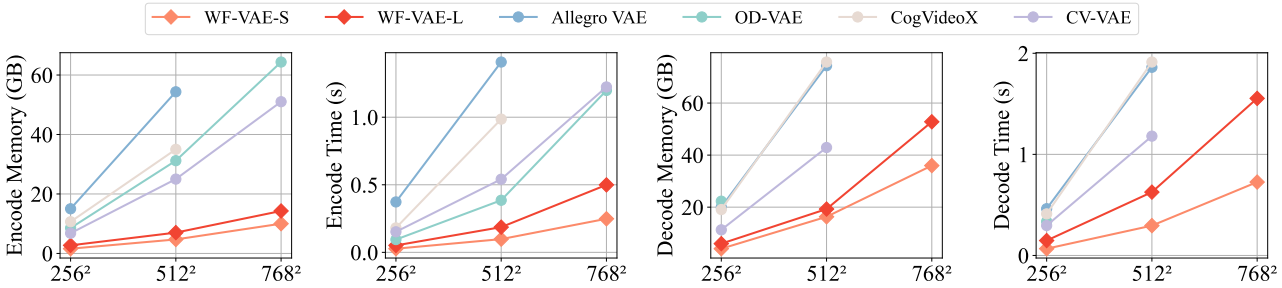


Figure 4. **Computational performance of encoding and decoding**. We evaluate the encoding, decoding time, and memory consumption across 33 frames with 256×256, 512×512, and 768×768 resolutions (benchmark models without causal convolution are tested with 32 frames). WF-VAE surpasses other VAE models by a large margin in terms of both inference speed and memory efficiency.

clips of 2,048 videos to compute FVD_{16}. Additionally, we evaluate the Inception Score (IS) [30] exclusively on the UCF-101 dataset, as suggested by [22]. We select Latte-L [22] as the denoiser. Since we focus not on the generative performance but on whether the latent spaces of various video VAEs facilitate practical diffusion model training, we chose not to use the higher-performing Latte-XL.

Training Strategy. We employ the AdamW [17, 21] optimizer with parameters $\beta_1 = 0.9$ and $\beta_2 = 0.999$, and set a fixed learning rate of 1×10^{-5}. Our training process comprises three stages: **(I)** the first stage aligns with [6], where we preprocess videos to 25 frames with 256×256 resolution, and a total batch size of 8. **(II)** we refresh the discriminator, increase the number of frames to 49, and reduce the FPS by half to enhance motion dynamics. **(III)** we observe that a large λ_{lpips} significantly affects video stability; therefore, we refresh the discriminator once more and set λ_{lpips} to 0.1. All three stages employ L1 loss: the initial stage is trained for 800,000 steps, while the subsequent stages are each trained for 200,000 steps. The training process utilizes 8 NVIDIA H100 GPUs. We implement a 3D discriminator and initiate GAN training from the start. All training hyperparameters are detailed in the appendix.

4.2. Comparison With Baseline Models

We compare WF-VAE with baseline models in three key aspects: computational efficiency, reconstruction performance, and diffusion-based generation performance. To ensure fairness in comparing metrics and model efficiency, we disable block-wise inference strategies across all VAEs.

Computational Efficiency. The computational efficiency evaluations are conducted using an H100 GPU with float32 precision. Performance evaluations are performed at 33 frames across multiple input resolutions. Due to non-causal convolution architecture, Allegro VAE is evaluated using 32-frame videos to maintain consistency. All benchmark VAEs use direct inference without block-wise inference strategies for a fair comparison. As shown in Fig. 4, WF-VAE demonstrates superior inference performance compared to other VAEs. For instance, WF-VAE-L requires 7170 MB of memory for encoding a video at 512×512 resolution, whereas OD-VAE demands approximately 31944 MB (445% higher). Similarly, CogVideoX consumes around 35849.33 MB (499% higher), and Allegro VAE requires 55664 MB (776% higher). These results highlight WF-VAE's significant advantages in large-scale train-

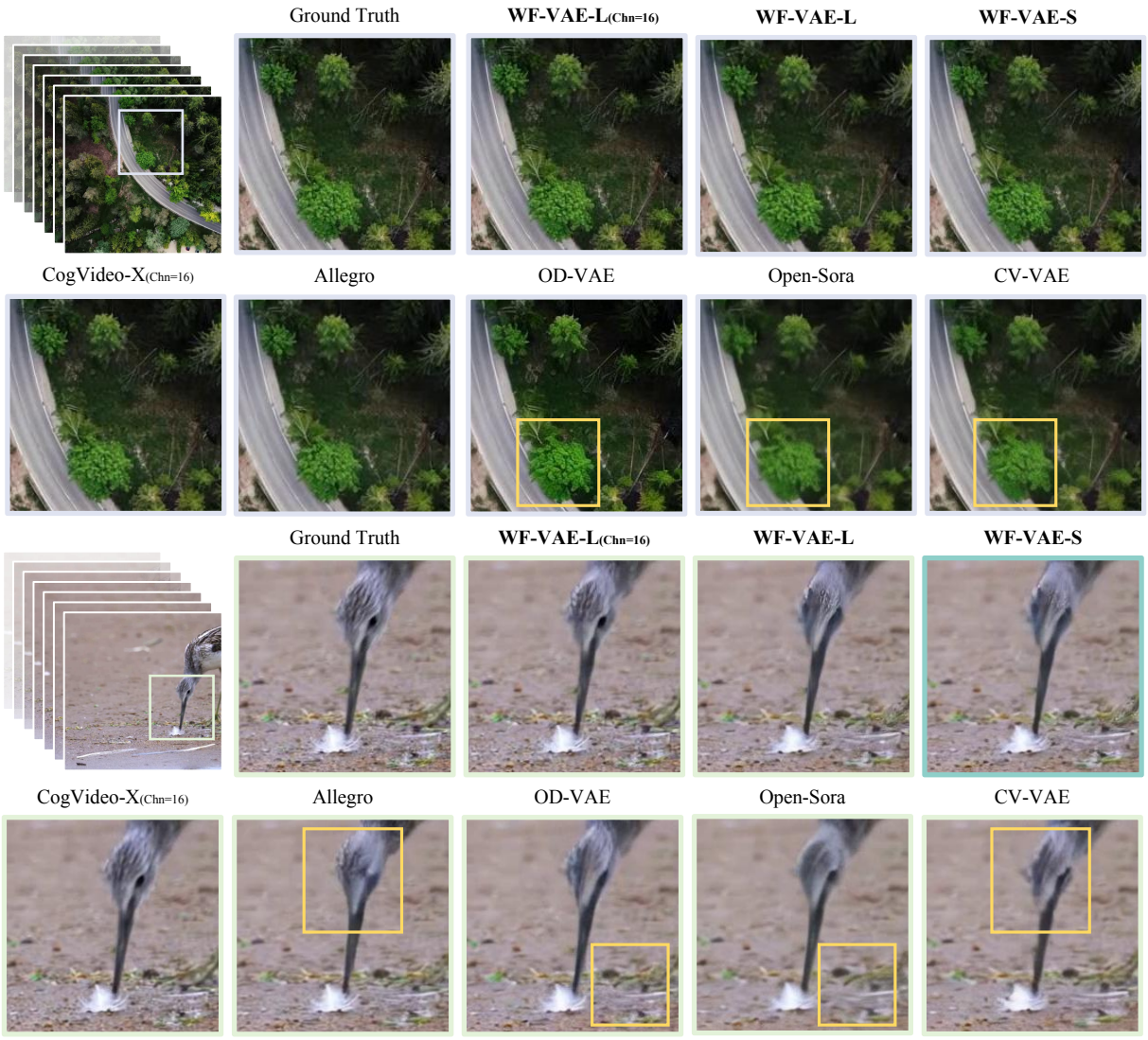


Figure 5. **Qualitative comparison of reconstruction performance.** We select two scenarios to comprehensively evaluate the visual quality of videos reconstructed by existing VAEs. Top: scenario contains **rich details**. Bottom: scenario contains **fast motion**.

ing and data processing. In terms of encoding speed, WF-VAE-L achieves an encoding time of 0.0513 seconds, while OD-VAE, CogVideoX, and Allegro VAE exhibit encoding times of 0.0945 seconds, 0.1810 seconds, and 0.3731 seconds, respectively, which are approximately 184%, 352%, and 727% slower. As illustrated in Fig. 4, WF-VAE demonstrates notable advantages in computational efficiency during the decoding process.

Video Reconstruction Performance. We present a quantitative comparison of reconstruction performance between WF-VAE and baseline models in Tab. 1 and qualitative reconstruction results in Fig. 5. Despite having the lowest computational cost, WF-VAE-S outperforms popular open-source video VAEs such as OD-VAE [6] and Open-Sora VAE [49] on both datasets. When increasing the model complexity, WF-VAE-L competes well with Allegro [50], outperforming it in PSNR, LPIPS and FVD but slightly lagging in SSIM. However, WF-VAE-L is significantly more computationally efficient than Allegro. Additionally, we compare WF-VAE-L with CogVideoX [40] using 16 latent channels. Except for a slightly lower SSIM on the Webvid-10M, WF-VAE-L outperforms CogVideoX across all other metrics. These results indicate that WF-VAE achieves competitive reconstruction performance compared to state-of-the-art open-source video VAEs, while substantially improving computational efficiency.

Video Generation Evaluation. Fig. 6 and Tab. 2 qualitatively and quantitatively demonstrate the video generation results of the diffusion model using WF-VAE. WF-VAE achieves the best performance in terms of FVD and IS

Method	Chn	SkyTimelapse	UCF101	
		$FVD_{16}\downarrow$	$FVD_{16}\downarrow$	IS↑
Allegro [50]	4	117.28	1045.66	67.16
OD-VAE [6]	4	130.79	1109.87	58.48
WF-VAE-S (Ours)	4	**103.44**	1005.10	65.89
WF-VAE-L (Ours)	4	113.67	**929.55**	**70.53**
CogVideoX [40]	16	109.20	1117.57	57.47
WF-VAE-L (Ours)	16	**108.69**	**947.18**	**71.86**

Table 2. **Quantitative evaluation of different VAE models for video generation.** We assess video generation quality using FVD_{16} on both SkyTimelapse and UCF-101 datasets, and IS on UCF-101 following prior work [22].

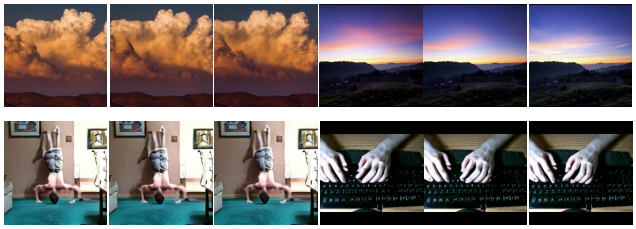
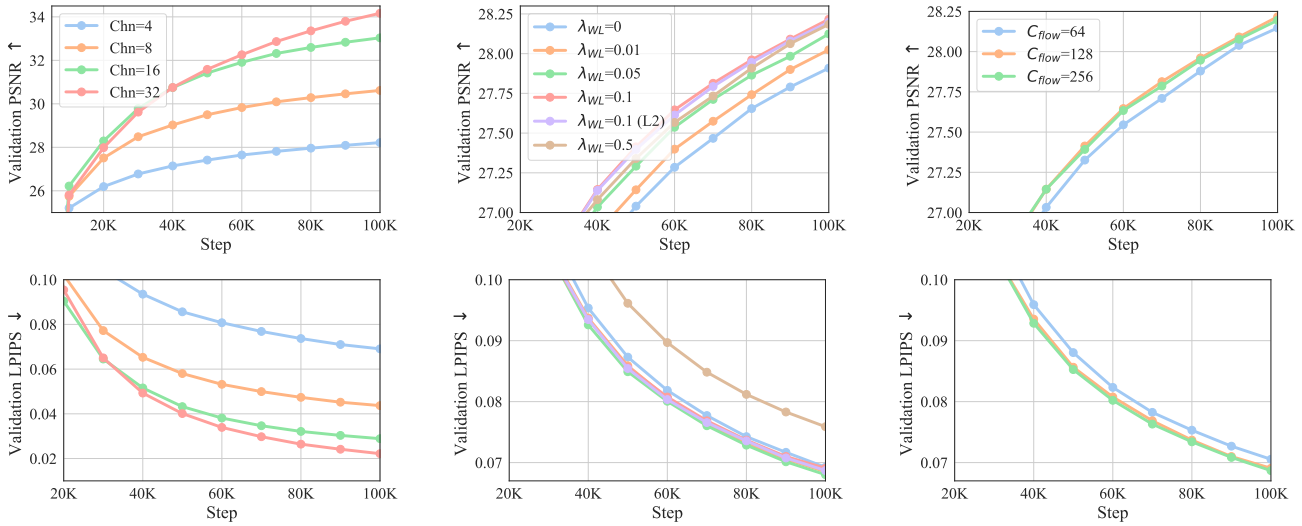


Figure 6. **Generated videos using WF-VAE with Latte-L. Top:** results trained with the SkyTimelapse dataset. **Bottom:** results trained with the UCF-101 dataset.



(a) Number of latent channels. (b) WL Loss weights λ_{WL}. (c) Number of energy flow path channels C_{flow}.

Figure 7. **Training dynamics under different settings.**

metrics. For instance, in the SkyTimelapse dataset, among models with 4 latent channels, WF-VAE-S achieves the best FVD score, 10.23 lower than WF-VAE-L and 27.35 lower than OD-VAE. For models with 16 latent channels, WF-VAE-L's FVD score is 0.51 lower than CogVideoX's. This might be because the higher dimensionality of the latent space makes convergence more difficult, resulting in slightly inferior performance compared to WF-VAE with 4 latent channels under the same training steps.

4.3. Ablation Study

Increasing the latent dimension. Recent works [8, 9] show that the number of latent channels significantly impacts reconstruction quality. We experiment with 4, 8, 16, and 32 latent channels. Fig. 7a illustrates that reconstruction performance improves significantly as the number of latent channels increases. However, larger latent channels may increase convergence difficulty in training diffusion model, as evidenced by the results presented in Tab. 2.

Exploration of WL loss weight λ_{WL}. To ensure structural symmetry in WF-VAE, we introduce WL loss. As shown in Fig. 7b, the model substantially decreases PSNR performance when $\lambda_{WL} = 0$. Our experiments demonstrate optimal results for both PSNR and LPIPS metrics when $\lambda_{WL} = 0.1$. Furthermore, our analysis of different loss functions reveals that utilizing L1 loss produces superior results compared to L2 loss.

Expanding the energy flow path. The low-frequency information flows into the latent representation through the energy flow pathway, and the number of channels in this path, C_{flow}, determines the intensity of low-frequency information injection into the backbone. We conducted experiments with C_{flow} values of 64, 128, and 256. As shown in Fig. 7c, we find that when C_{flow} is 128, it can balance reconstruction and computational performance.

Increasing the number of base channels. To further exploit the capabilities of the WF-VAE architecture, we increase the model complexity by expanding the number of

Model	BC	Params (M)		Kinetics-400	
		Enc	Dec	PSNR↑	LPIPS↓
WF-VAE-S	128	38	108	28.21	0.0779
WF-VAE-M	160	58	164	28.44	0.0699
WF-VAE-L	192	84	232	**28.66**	**0.0661**

Table 3. **Scalability of WF-VAE.** We evaluated PSNR and LPIPS on Kinetics-400 [16]. Reconstruction performance improves as model complexity increases.

base channels. The channel dimensionality increases by one base channel width after each downsampling layer, starting from the number of base channels. As shown in Tab. 3, we experiment with three configurations: 128, 160, and 192 base channels. The results demonstrate that the model's performance improves as the number of base channels increases. Despite the corresponding rise in model parameters, the computational cost remains comparatively low compared to benchmark models, as shown in Fig. 4.

Ablation studies of model architecture. First, We evaluate the effectiveness of the energy pathway by examining its impact when removed from layer 3 alone and both layers 2 and 3. Specifically, we eliminate the wavelet transform layer and inflow-outflow module, setting the `concat` input to a zero tensor. This analysis highlights the advantages of integrating low-frequency video energy into the backbone. Second, we investigate the importance of the proposed WL Loss in regularizing the encoder-decoder. Third, we analyze the effect of replacing the normalization method with layer normalization [1] for *Causal Cache*. The results of these ablation studies are shown in Tab. 4.

Settings					Kinetics-400	
L1	L2	L3	WL Loss	NM	PSNR↑	LPIPS↓
✓				L	27.85	0.0737
✓	✓		✓	L	27.94	0.0737
✓	✓	✓		L	27.90	0.0692
✓	✓	✓	✓	L	**28.21**	0.0690
✓	✓	✓	✓	G	28.03	**0.0684**

Table 4. **Ablation studies on model architecture.** We evaluate the impact of three key components: energy flow pathways across network layers, WL loss, and normalization methods (L: layer normalization [1], G: group normalization [37]).

4.4. Causal Cache

To validate the lossless inference capability of *Causal Cache*, we compare our approach with existing block-wise inference methods implemented in several open-source LVDMs. OD-VAE [6] and Allegro [50] offer spatio-temporal tiling inference implementations, while CogVideoX [40] adopts a temporal caching strategy. As demonstrated in Tab. 5, both tiling strategies and conventional caching methods exhibited performance degradation, while *Causal Cache* achieves lossless inference with performance metrics identical to direct inference.

Method	Chn	BWI	Panda70M	
			PSNR↑	LPIPS↓
Allegro [50]	4	✗	31.71	0.0422
		✓	25.31(-6.40)	0.1124(+0.0702)
OD-VAE [6]	4	✗	30.31	0.0439
		✓	28.51(-1.80)	0.0552(+0.0113)
WF-VAE-L (Ours)	4	✗	32.10	0.0411
		✓	32.10(0.00)	0.0411(0.0000)
CogVideoX [40]	16	✗	35.79	0.0198
		✓	35.41(-0.38)	0.0218(+0.0020)
WF-VAE-L (Ours)	16	✗	35.87	0.0175
		✓	35.87(0.00)	0.0175(0.0000)

Table 5. **Quantitative analysis of visual quality degradation induced by block-wise inference.** Values in red indicate degradation compared to direct inference, while values in green demonstrate preservation of quality. BWI denotes Block-Wise Inference. Experiments are conducted on 33 frames with 256×256 resolution.

5. Conclusion

In this paper, we propose **WF-VAE**, an innovative autoencoder that utilizes multi-level wavelet transform to extract pyramidal features, thus creating a primary energy flow pathway for encoding low-frequency video information into a latent representation. Additionally, we introduce a lossless block-wise inference mechanism called *Causal Cache*, which completely resolves video flickering associated with prior tiling strategies. Our experiments demonstrate that WF-VAE achieves state-of-the-art reconstruction performance while maintaining low computational costs. WF-VAE significantly reduces the expenses associated with large-scale video pre-training, potentially inspiring future designs of video VAEs.

Limitations and Future Work. The initial design of the decoder incorporated insights from [27], employing a highly complex structure that resulted in more parameters in the backbone of the decoder compared to the encoder. Although the computational cost remains manageable, we consider these parameters redundant. Consequently, we aim to streamline the model in future work to fully leverage the advantages of our architecture.

Acknowledgments

We thank all the anonymous reviewers for their constructive comments. This work was supported in part by the Natural Science Foundation of China (No. 62202014, 62332002, 62425101, 62088102).

References

[1] Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton. Layer normalization. *arXiv preprint arXiv:1607.06450*, 2016. 8

[2] Max Bain, Arsha Nagrani, Gul Varol, and Andrew Zisserman. Frozen in time: A joint video and image encoder for end-to-end retrieval. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, Oct 2021. 4, 5

[3] Fan Bao, Chendong Xiang, Gang Yue, Guande He, Hongzhou Zhu, Kaiwen Zheng, Min Zhao, Shilong Liu, Yaole Wang, and Jun Zhu. Vidu: a highly consistent, dynamic and skilled text-to-video generator with diffusion models. *arXiv preprint arXiv:2405.04233*, 2024. 2

[4] Andreas Blattmann, Tim Dockhorn, Sumith Kulal, Daniel Mendelevitch, Maciej Kilian, Dominik Lorenz, Yam Levi, Zion English, Vikram Voleti, Adam Letts, et al. Stable video diffusion: Scaling latent video diffusion models to large datasets. *arXiv preprint arXiv:2311.15127*, 2023. 4, 5

[5] Tim Brooks, Bill Peebles, Connor Holmes, Will DePue, Yufei Guo, Li Jing, David Schnurr, Joe Taylor, Troy Luhman, Eric Luhman, et al. Video generation models as world simulators. *OpenAI Blog*, 1:8, 2024. 1

[6] Liuhan Chen, Zongjian Li, Bin Lin, Bin Zhu, Qian Wang, Shenghai Yuan, Xing Zhou, Xinghua Cheng, and Li Yuan. Od-vae: An omni-dimensional video compressor for improving latent video diffusion model. *arXiv preprint arXiv:2409.01199*, 2024. 1, 2, 4, 5, 6, 7, 8

[7] Tsai-Shien Chen, Aliaksandr Siarohin, Willi Menapace, Ekaterina Deyneka, Hsiang-wei Chao, Byung Eun Jeon, Yuwei Fang, Hsin-Ying Lee, Jian Ren, Ming-Hsuan Yang, and Sergey Tulyakov. Panda-70m: Captioning 70m videos with multiple cross-modality teachers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 13320–13331, June 2024. 4, 5

[8] Xiaoliang Dai, Ji Hou, Chih-Yao Ma, Sam Tsai, Jialiang Wang, Rui Wang, Peizhao Zhang, Simon Vandenhende, Xiaofang Wang, Abhimanyu Dubey, et al. Emu: Enhancing image generation models using photogenic needles in a haystack. *arXiv preprint arXiv:2309.15807*, 2023. 7

[9] Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, et al. Scaling rectified flow transformers for high-resolution image synthesis. In *Forty-first international conference on machine learning*, 2024. 2, 7

[10] Patrick Esser, Robin Rombach, and Bjorn Ommer. Taming transformers for high-resolution image synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 12873–12883, June 2021. 4

[11] Shahaf E Finder, Roy Amoyal, Eran Treister, and Oren Freifeld. Wavelet convolutions for large receptive fields. In *European Conference on Computer Vision*, pages 363–380. Springer, 2024. 2

[12] Rinon Gal, Dana Cohen Hochberg, Amit Bermano, and Daniel Cohen-Or. Swagan: A style-based wavelet-driven generative model. *ACM Transactions on Graphics (TOG)*, 40(4):1–11, 2021. 2

[13] Yuwei Guo, Ceyuan Yang, Anyi Rao, Zhengyang Liang, Yaohui Wang, Yu Qiao, Maneesh Agrawala, Dahua Lin, and Bo Dai. Animatediff: Animate your personalized text-to-image diffusion models without specific tuning. *International Conference on Learning Representations*, 2024. 2

[14] Alain Hore and Djemel Ziou. Image quality metrics: Psnr vs. ssim. In *2010 20th international conference on pattern recognition*, pages 2366–2369. IEEE, 2010. 4

[15] Huaibo Huang, Ran He, Zhenan Sun, and Tieniu Tan. Wavelet-srnet: A wavelet-based cnn for multi-scale face super resolution. In *Proceedings of the IEEE international conference on computer vision*, pages 1689–1697, 2017. 2

[16] Will Kay, Joao Carreira, Karen Simonyan, Brian Zhang, Chloe Hillier, Sudheendra Vijayanarasimhan, Fabio Viola, Tim Green, Trevor Back, Paul Natsev, Mustafa Suleyman, and Andrew Zisserman. The kinetics human action video dataset, 2017. 4, 8

[17] DiederikP. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv: Learning,arXiv: Learning*, Dec 2014. 5

[18] DiederikP. Kingma and Max Welling. Auto-encoding variational bayes. *arXiv: Machine Learning,arXiv: Machine Learning*, Dec 2013. 2, 4

[19] Bin Lin, Yunyang Ge, Xinhua Cheng, Zongjian Li, Bin Zhu, Shaodong Wang, Xianyi He, Yang Ye, Shenghai Yuan, Liuhan Chen, et al. Open-sora plan: Open-source large video generation model. *arXiv preprint arXiv:2412.00131*, 2024. 1, 2, 4

[20] Bin Lin, Yang Ye, Bin Zhu, Jiaxi Cui, Munan Ning, Peng Jin, and Li Yuan. Video-llava: Learning united visual representation by alignment before projection. *arXiv preprint arXiv:2311.10122*, 2023. 2

[21] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017. 5

[22] Xin Ma, Yaohui Wang, Gengyun Jia, Xinyuan Chen, Ziwei Liu, Yuan-Fang Li, Cunjian Chen, and Yu Qiao. Latte: Latent diffusion transformer for video generation. *arXiv preprint arXiv:2401.03048*, 2024. 4, 5, 7

[23] S.G. Mallat. A theory for multiresolution signal decomposition: the wavelet representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, page 674–693, Jun 1989. 2

[24] Gautam Mittal, Jesse Engel, Curtis Hawthorne, and Ian Simon. Symbolic music generation with diffusion models. *arXiv: Sound,arXiv: Sound*, Mar 2021. 2

[25] Adam Polyak, Amit Zohar, Andrew Brown, Andros Tjandra, Animesh Sinha, Ann Lee, Apoorv Vyas, Bowen Shi, Chih-Yao Ma, Ching-Yao Chuang, David Yan, Dhruv Choudhary, Dingkang Wang, Geet Sethi, Guan Pang, Haoyu Ma, Ishan Misra, Ji Hou, Jialiang Wang, Kiran Jagadeesh, Kunpeng Li, Luxin Zhang, Mannat Singh, Mary Williamson, Matt Le, Matthew Yu, Mitesh Kumar Singh, Peizhao Zhang, Peter Vajda, Quentin Duval, Rohit Girdhar, Roshan Sumbaly, Sai Saketh Rambhatla, Sam Tsai, Samaneh Azadi, Samyak Datta, Sanyuan Chen, Sean Bell, Sharadh Ramaswamy, Shelly Sheynin, Siddharth Bhattacharya, Simran Motwani, Tao Xu, Tianhe Li, Tingbo Hou, Wei-Ning Hsu, Xi Yin, Xiaoliang Dai, Yaniv Taigman, Yaqiao Luo, Yen-Cheng Liu, Yi-Chiao Wu, Yue Zhao, Yuval Kirstain, Zecheng He, Zijian

He, Albert Pumarola, Ali Thabet, Artsiom Sanakoyeu, Arun Mallya, Baishan Guo, Boris Araya, Breena Kerr, Carleigh Wood, Ce Liu, Cen Peng, Dimitry Vengertsev, Edgar Schonfeld, Elliot Blanchard, Felix Juefei-Xu, Fraylie Nord, Jeff Liang, John Hoffman, Jonas Kohler, Kaolin Fire, Karthik Sivakumar, Lawrence Chen, Licheng Yu, Luya Gao, Markos Georgopoulos, Rashel Moritz, Sara K. Sampson, Shikai Li, Simone Parmeggiani, Steve Fine, Tara Fowler, Vladan Petrovic, and Yuming Du. Movie gen: A cast of media foundation models, 2024. 1, 2

[26] Aleš Procházka, Lucie Gráfová, Oldrich Vyšata, and Neurocenter Caregroup. Three-dimensional wavelet transform in multi-dimensional biomedical volume processing. In *Proc. of the IASTED International Conference on Graphics and Virtual Reality, Cambridge*, volume 263, page 268, 2011. 2

[27] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10684–10695, 2022. 1, 2, 4, 5, 8

[28] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *Medical image computing and computer-assisted intervention–MICCAI 2015: 18th international conference, Munich, Germany, October 5-9, 2015, proceedings, part III 18*, pages 234–241. Springer, 2015. 2

[29] Seyedmorteza Sadat, Jakob Buhmann, Derek Bradley, Otmar Hilliges, and Romann M Weber. Litevae: Lightweight and efficient variational autoencoders for latent diffusion models. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. 2

[30] Masaki Saito, Eiichi Matsumoto, and Shunta Saito. Temporal generative adversarial nets with singular value clipping. In *2017 IEEE International Conference on Computer Vision (ICCV)*, Oct 2017. 5

[31] Abhishek Sinha, Jiaming Song, Chenlin Meng, and Stefano Ermon. D2c: Diffusion-denoising models for few-shot conditional generation. *arXiv: Learning,arXiv: Learning*, Jun 2021. 2

[32] Ivan Skorokhodov, Sergey Tulyakov, and Mohamed Elhoseiny. Stylegan-v: A continuous video generator with the price, image quality and perks of stylegan2. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 3626–3636, 2022. 4

[33] Khurram Soomro, Amir Zamir, and Mubarak Shah. Ucf101: A dataset of 101 human actions classes from videos in the wild. *arXiv: Computer Vision and Pattern Recognition,arXiv: Computer Vision and Pattern Recognition*, Dec 2012. 4

[34] Thomas Unterthiner, Sjoerdvan Steenkiste, Karol Kurach, Raphaël Marinier, Marcin Michalski, and Sylvain Gelly. Fvd: A new metric for video generation. *International Conference on Learning Representations,International Conference on Learning Representations*, Mar 2019. 4

[35] Arash Vahdat, Karsten Kreis, and Jan Kautz. Score-based generative modeling in latent space. *Neural Information Processing Systems,Neural Information Processing Systems*, Dec 2021. 2

[36] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004. 4

[37] Yuxin Wu and Kaiming He. Group normalization. In *Proceedings of the European conference on computer vision (ECCV)*, pages 3–19, 2018. 2, 8

[38] Wei Xiong, Wenhan Luo, Lin Ma, Wei Liu, and Jiebo Luo. Learning to generate time-lapse videos using multi-stage dynamic generative adversarial networks. *Cornell University - arXiv,Cornell University - arXiv*, Sep 2017. 4

[39] Jiaqi Xu, Xinyi Zou, Kunzhe Huang, Yunkuo Chen, Bo Liu, MengLi Cheng, Xing Shi, and Jun Huang. Easyanimate: A high-performance long video generation method based on transformer architecture. *arXiv preprint arXiv:2405.18991*, 2024. 1, 2

[40] Zhuoyi Yang, Jiayan Teng, Wendi Zheng, Ming Ding, Shiyu Huang, Jiazheng Xu, Yuanming Yang, Wenyi Hong, Xiaohan Zhang, Guanyu Feng, et al. Cogvideox: Text-to-video diffusion models with an expert transformer. *arXiv preprint arXiv:2408.06072*, 2024. 1, 2, 4, 5, 6, 7, 8

[41] Lijun Yu, José Lezama, Nitesh Bharadwaj Gundavarapu, Luca Versari, Kihyuk Sohn, David Minnen, Yong Cheng, Agrim Gupta, Xiuye Gu, Alexander G Hauptmann, et al. Language model beats diffusion-tokenizer is key to visual generation. In *ICLR*, 2024. 2, 4

[42] Shenghai Yuan, Jijia Chen, Wenchao Jiang, Zhiming Zhao, and Song Guo. Lhnetv2: A balanced low-cost hybrid network for single image dehazing. *IEEE Transactions on Multimedia*, 2024. 2

[43] Shenghai Yuan, Jijia Chen, Jiaqi Li, Wenchao Jiang, and Song Guo. Lhnet: A low-cost hybrid network for single image dehazing. In *Proceedings of the 31st ACM International Conference on Multimedia*, pages 7706–7717, 2023. 2

[44] Shenghai Yuan, Jinfa Huang, Xianyi He, Yunyuan Ge, Yujun Shi, Liuhan Chen, Jiebo Luo, and Li Yuan. Identity-preserving text-to-video generation by frequency decomposition. *arXiv preprint arXiv:2411.17440*, 2024. 1

[45] Shenghai Yuan, Jinfa Huang, Yujun Shi, Yongqi Xu, Ruijie Zhu, Bin Lin, Xinhua Cheng, Li Yuan, and Jiebo Luo. Magictime: Time-lapse video generation models as metamorphic simulators. *arXiv preprint arXiv:2404.05014*, 2024. 2

[46] Shenghai Yuan, Jinfa Huang, Yongqi Xu, Yaoyang Liu, Shaofeng Zhang, Yujun Shi, Rui-Jie Zhu, Xinhua Cheng, Jiebo Luo, and Li Yuan. Chronomagic-bench: A benchmark for metamorphic evaluation of text-to-time-lapse video generation. *Advances in Neural Information Processing Systems*, 37:21236–21270, 2024. 2

[47] Richard Zhang, Phillip Isola, Alexei A. Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Jun 2018. 4

[48] Sijie Zhao, Yong Zhang, Xiaodong Cun, Shaoshu Yang, Muyao Niu, Xiaoyu Li, Wenbo Hu, and Ying Shan. Cv-vae: A compatible video vae for latent generative video models. *arXiv preprint arXiv:2405.20279*, 2024. 1, 2, 4, 5

[49] 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. 1, 2, 4, 5, 6

[50] Yuan Zhou, Qiuyue Wang, Yuxuan Cai, and Huan Yang. Allegro: Open the black box of commercial-level video generation model. *arXiv preprint arXiv:2410.15458*, 2024. 1, 2, 4, 5, 6, 7, 8

[51] Bin Zhu, Bin Lin, Munan Ning, Yang Yan, Jiaxi Cui, HongFa Wang, Yatian Pang, Wenhao Jiang, Junwu Zhang, Zongwei Li, et al. Languagebind: Extending video-language pretraining to n-modality by language-based semantic alignment. *arXiv preprint arXiv:2310.01852*, 2023. 2