From Slow Bidirectional to Fast Autoregressive Video Diffusion Models

Tianwei Yin[1*] Qiang Zhang[2*] Richard Zhang[2]
William T. Freeman[1] Frédo Durand[1] Eli Shechtman[2] Xun Huang[2]
[1]MIT [2]Adobe

https://causvid.github.io/

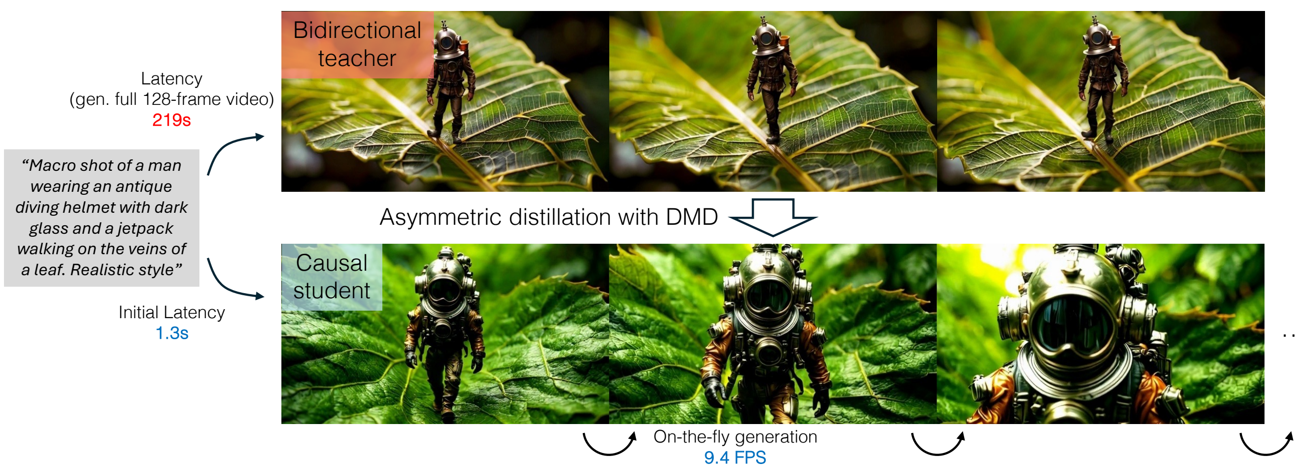


Figure 1. Traditional bidirectional diffusion models (top) deliver high-quality outputs but suffer from significant latency, taking 219 seconds to generate a 128-frame video. Users must wait for the entire sequence to complete before viewing any results. In contrast, we distill the bidirectional diffusion model into a few-step autoregressive generator (bottom), dramatically reducing computational overhead. Our model (**CausVid**) achieves an initial latency of only 1.3 seconds, after which frames are generated continuously in a streaming fashion at approximately 9.4 FPS, facilitating interactive workflows for video content creation.

Abstract

Current video diffusion models achieve impressive generation quality but struggle in interactive applications due to bidirectional attention dependencies. The generation of a single frame requires the model to process the entire sequence, including the future. We address this limitation by adapting a pretrained bidirectional diffusion transformer to an autoregressive transformer that generates frames on-the-fly. To further reduce latency, we extend distribution matching distillation (DMD) to videos, distilling 50-step diffusion model into a 4-step generator. To enable stable and high-quality distillation, we introduce a student initialization scheme based on teacher's ODE trajectories, as well as an asymmetric distillation strategy that supervises a causal student model with a bidirectional teacher. This approach effectively mitigates error accumulation in autoregressive generation, allowing long-duration video synthesis despite training on short clips. Our model achieves a total score of 84.27 on the VBench-Long benchmark, surpassing all previous video generation models. It enables fast streaming generation of high-quality videos at 9.4 FPS on a single GPU thanks to KV caching. Our approach also enables streaming video-to-video translation, image-to-video, and dynamic prompting in a zero-shot manner. We will release the code based on an open-source model in the future.

1. Introduction

The emergence of diffusion models has revolutionized how we can create videos from text [3, 5, 25, 29, 63, 98, 111]. Many of the state-of-the-art video diffusion models rely on the Diffusion Transformer (DiT) architecture [2, 61], which usually employs bidirectional attention across all video frames. Despite the impressive quality, the bidirec-

[*] Equal Contribution. Correspondence: tianweiy@mit.edu

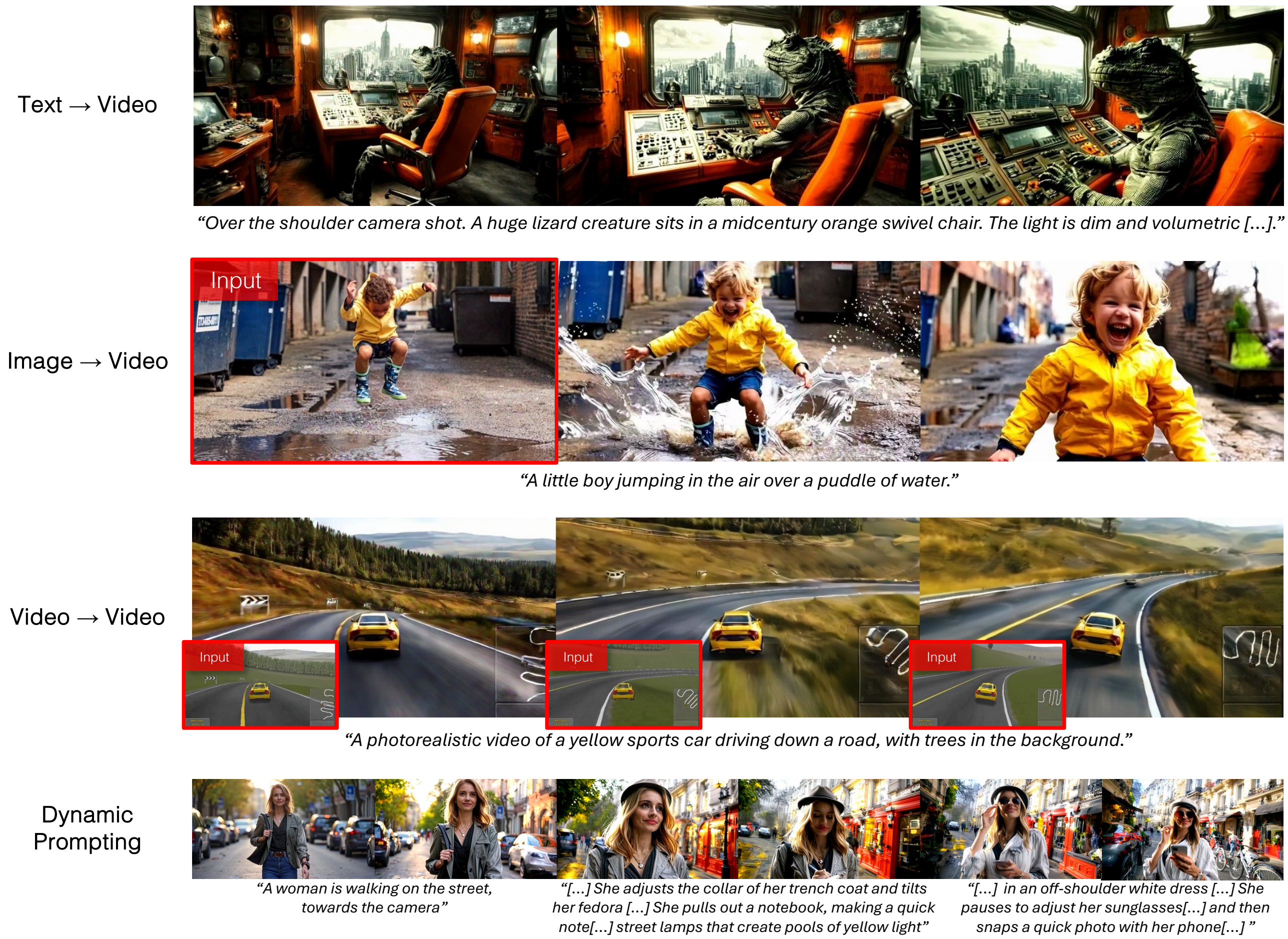


Figure 2. Diverse video generation tasks supported by our method. The model can generate videos from a single text prompt (top row) or with additional image input (second row). Our model also enables interactive applications where generation results respond to user input with low latency. For example, it can add realistic texture and lighting to outputs rendered by a basic game engine that responds to user input on the fly (third row). Additionally, it enables dynamic prompting (fourth row), allowing users to input new prompts at any point in a video to build extended narratives with evolving actions and environments.

tional dependencies imply that generating a single frame requires processing the entire video. This introduces long latency and prevents the model from being applied to interactive and streaming applications, where the model needs to continually generate frames based on user inputs that may change over time. The generation of the current frame depends on future conditional inputs that are not yet available. Current video diffusion models are also limited by their speed. The compute and memory costs increase quadratically with the number of frames, which, combined with the large number of denoising steps during inference, makes generating long videos prohibitively slow and expensive.

Autoregressive models offer a promising solution to address some of these limitations, but they face challenges with error accumulation and computational efficiency. Instead of generating all frames simultaneously, autoregressive video models generate frames sequentially. Users can start watching the video as soon as the first frame is generated, without waiting for the entire video to be completed. This reduces latency, removes limitations on video duration, and opens the door for interactive control. However, autoregressive models are prone to error accumulation: each generated frame builds on potentially flawed previous frames, causing prediction errors to magnify and worsen over time. Moreover, although the latency is reduced, existing autoregressive video models are still far from being able to generate realistic videos at interactive frame rate [7, 29, 37].

In this paper, we introduce **CausVid**, a model designed for fast and interactive **caus**al **vid**eo generation. We design an autoregressive diffusion transformer architecture with causal dependencies between video frames. Similar to the popular decoder-only large language models (LLMs) [6,

Figure 3. Our model, CausVid, demonstrates that autoregressive video diffusion can be effectively scaled up for general text-to-video tasks, achieving quality on par with bidirectional diffusion models. Moreover, when combined with distillation techniques, it delivers multiple orders of magnitude speedup. Please visit our website for more visualizations.

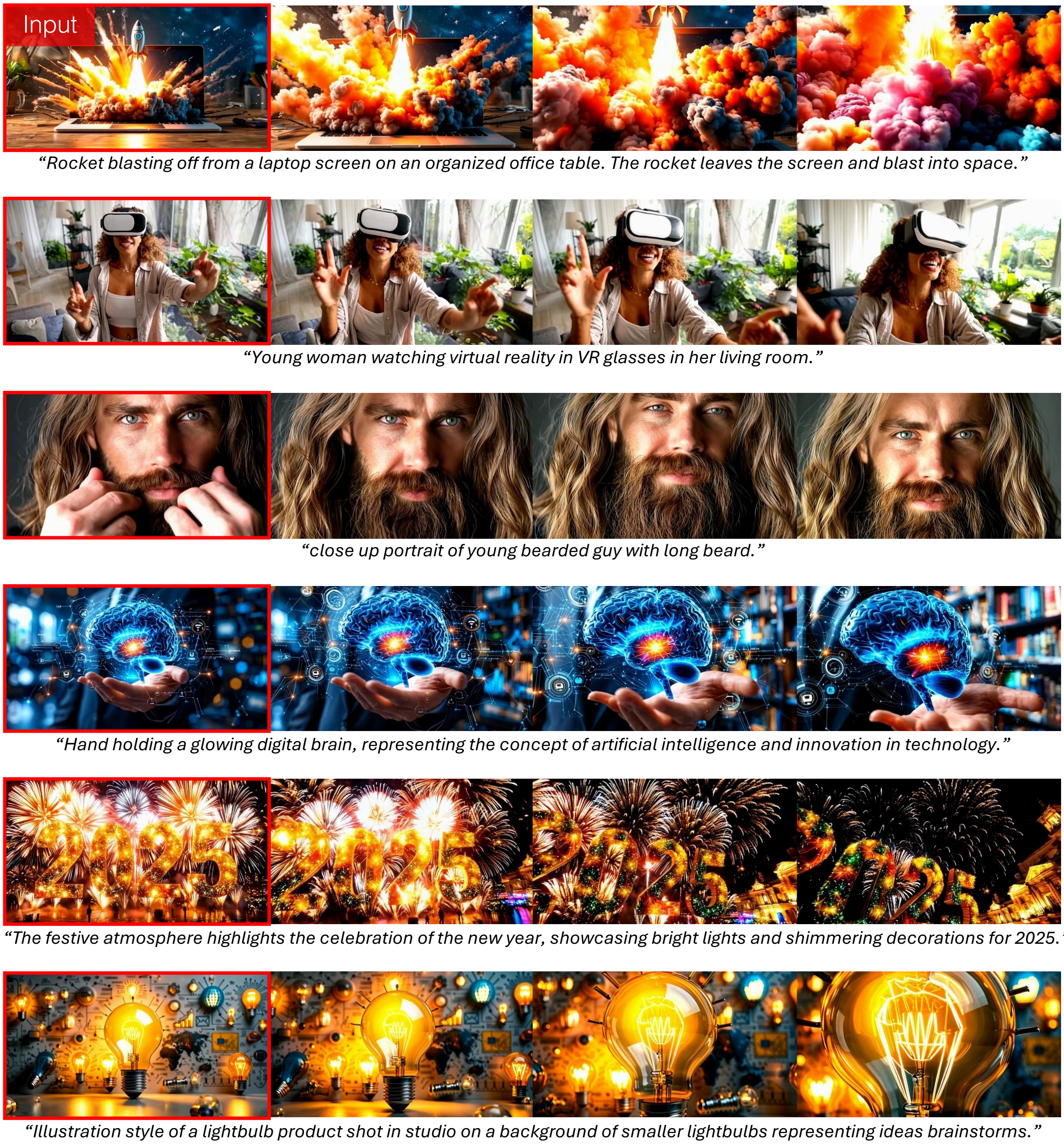


Figure 4. Trained exclusively on text-to-video generation, our model, CausVid, can be applied zero-shot to image-to-video tasks thanks to its autoregressive design. In the examples shown, the first column represents the input image, while the subsequent frames are generated outputs. Please visit our website for more visualizations.

66], our model achieves sample-efficient training by leveraging supervision from all input frames at each iteration, as well as efficient autoregressive inference through key-value (KV) caching. To further improve generation speed, we adapt distribution matching distillation (DMD) [101, 102], a few-step distillation approach originally designed for image diffusion models, to video data. Instead of naively distilling an autoregressive diffusion model [8, 29] into a few-step student, we propose an *asymmetric distillation* strategy where we distill the knowledge in a pretrained teacher diffusion model with *bidirectional* attention into our *causal* student model. We show that this asymmetric distillation approach significantly reduced error accumulation during autoregressive inference. This allows us to support autoregressively generating videos that are much longer than the ones seen during training. Comprehensive experiments demonstrate that our model achieves video quality on par with state-of-the-art bidirectional diffusion models while offering enhanced interactivity and speed. To our knowledge, this is the first autoregressive video generation method that competes with bidirectional diffusion in terms of quality (Fig. 3 and Fig. 4). Additionally, we showcase the versatility of our method in tasks such as image-to-video generation, video-to-video translation, and dynamic prompting, all achievable with extremely low latency (Fig. 2).

2. Related Work

Autoregressive Video Generation. Given the inherent temporal order of video data, it is intuitively appealing to model video generation as an autoregressive process. Early research uses either regression loss [20, 49] or GAN loss [38, 58, 80, 83] to supervise the frame prediction task. Inspired by the success of LLMs [6], some works choose to tokenize video frames into discrete tokens and apply autoregressive transformers to generate tokens one by one [18, 37, 43, 88, 92, 97]. However, this approach is computationally expensive as each frame usually consists of thousands of tokens. Recently, diffusion models have emerged as a promising approach for video generation. While most video diffusion models have bidirectional dependencies [5, 63, 98, 111], autoregressive video generation using diffusion models has also been explored. Some works [1, 29, 81, 106] train video diffusion models to denoise new frames given context frames. Others [8, 34, 70] train the model to denoise the entire video under the setting where different frames may have different noise levels. Therefore, they support autoregressive sampling as a special case where the current frame is noisier than previous ones. A number of works have explored adapting pretrained text-to-image [36, 42, 81, 90] or text-to-video [17, 23, 34, 93, 95] diffusion models to be conditioned on context frames, enabling autoregressive video generation. Our method is closely related to this line of work, with the difference that we introduce a novel adaption method through diffusion distillation, significantly improving efficiency and making autoregressive methods competitive with bidirectional diffusion for video generation.

Long Video Generation. Generating long and variable-length videos remains a challenging task. Some works [12, 65, 79, 84, 85, 104, 108] generate multiple overlapped clips simultaneously using video diffusion model pretrained on fixed and limited-length clips, while employing various techniques to ensure temporal coherence. Another approach is to generate long videos hierarchically, first generating sparse keyframes and then interpolating between them [100, 109]. Unlike full-video diffusion models that are trained to generate videos of fixed length, autoregressive models [17, 23, 29, 43, 88, 93] are inherently suitable for generating videos of various length, although they may suffer from error accumulation when generating long sequences. We find that the distribution matching objective with a bidirectional teacher is surprisingly helpful for reducing the accumulation of errors, enabling both efficient and high-quality long video generation.

Diffusion Distillation. Diffusion models typically require many denoising steps to generate high-quality samples, which can be computationally expensive [24, 75]. Distillation techniques train a student model to generate samples in fewer steps by mimicking the behavior of a teacher diffusion model [30, 53, 60, 67, 71, 72, 78, 96, 102]. Luhman *et al*. [53] train a single-step student network to approximate the noise-image mapping obtained from a DDIM teacher model [75]. Progressive Distillation [71] trains a sequence of student models, reducing the number of steps by half at each stage. Consistency Distillation [22, 33, 54, 76, 78] trains the student to map any points on an ODE trajectory to its origin. Rectified flow [15, 47, 48] trains a student model on the linear interpolation path of noise-image pairs obtained from the teacher. Adversarial loss [19] is also used, sometimes in combination with other methods, to improve the quality of student output [30, 45, 72, 96, 101]. DMD [101, 102] optimizes an approximate reverse KL divergence [16, 55, 89, 99], whose gradients can be represented as the difference of two score functions trained on the data and generator's output distribution, respectively. Unlike trajectory-preserving methods [33, 48, 71], DMD provides supervision at the distribution level and offers the unique advantage of allowing different architectural formulations for the teacher and student diffusion models. Our approach builds upon the effectiveness and flexibility of DMD to train an autoregressive generator by distilling from a bidirectional teacher diffusion model.

Recently, researchers have begun to apply distillation methods to video diffusion models, such as progressive distillation [44], consistency distillation [40, 57, 86, 87], and adversarial distillation [57, 107]. Most approaches focus on distilling models designed to generate short videos (less than 2 seconds). Moreover, they focus on distilling a non-causal teacher into a student that is also non-causal. In contrast, our method distills a non-causal teacher into a causal student, enabling streaming video generation. Our generator is trained on 10-second videos and can generate infinitely long videos via sliding window inference. There has been another line of work that focuses on improving the efficiency of video diffusion models by system-level optimization (e.g., caching and parallelism) [46, 105, 110, 114]. However, they are usually applied to standard multi-step diffusion models and can be combined with our distillation approach, further improving the throughput and latency.

3. Background

This section provides background information on video diffusion models (Sec. 3.1) and distribution matching distillation (Sec. 3.2), which our method is built upon.

3.1. Video Diffusion Models

Diffusion models [24, 74] generate samples from a data distribution $p(x_0)$ by progressively denoising samples that are initially drawn from a Gaussian distribution $p(x_T)$. They are trained to denoise samples created by adding random noise ϵ to the samples x_0 from the data distribution

$$x_t = \alpha_t x_0 + \sigma_t \epsilon,\ \epsilon \sim \mathcal{N}(0, I)\,, \tag{1}$$

where $\alpha_t, \sigma_T > 0$ are scalars that jointly define the signal-to-noise ratio according to a specific noise schedule [31, 35, 77] at step t. The denoiser with parameter θ is typically trained to predict the noise [24]

$$\mathcal{L}(\theta) = \mathbb{E}_{t,x_0,\epsilon} \left\| \epsilon_\theta(x_t, t) - \epsilon \right\|_2^2\,. \tag{2}$$

Alternative prediction targets include the clean image x_0 [31, 71] or a weighted combination of x_0 and ϵ known as v-prediction [71]. All prediction schemes are fundamentally related to the score function, which represents the gradient of the log probability of the distribution [35, 77]:

$$s_\theta(x_t, t) = \nabla_{x_t} \log p(x_t) = -\frac{\epsilon_\theta(x_t, t)}{\sigma_t}. \tag{3}$$

In the following sections, we simplify our notation by using the score function s_θ as a general representation of the diffusion model, while noting that it can be derived through reparameterization from a pretrained model of any prediction scheme. At inference time, we start from full Gaussian noise x_T and progressively apply the diffusion model to generate a sequence of increasingly cleaner samples. There are many possible sampling methods [31, 52, 75, 103] to compute the sample x_{t-1} in the next time step from the current one x_t based on the predicted noise $\epsilon_\theta(x_t, t)$.

Diffusion models can be trained on either raw data [24, 28, 31] or on a lower-dimensional latent space obtained by a variational autoencoder (VAE) [32, 61, 68, 98, 111]. The latter is often referred to as latent diffusion models (LDMs) and has become the standard approach for modeling high-dimensional data such as videos [4, 26, 98, 111, 113]. The autoencoder usually compresses both spatial and temporal dimensions of the video, making diffusion models easier to learn. The denoiser network in video diffusion models can be instantiated by different neural architectures, such as U-Net [10, 25, 69, 113] or Transformers [5, 26, 82, 98].

3.2. Distribution Matching Distillation

Distribution matching distillation is a technique designed to distill a slow, multi-step teacher diffusion model into an efficient few-step student model [101, 102]. The core idea is to minimize the reverse KL divergence across randomly sampled timesteps t between the smoothed data distribution $p_{\text{data}}(x_t)$ and the student generator's output distribution $p_{\text{gen}}(x_t)$. The gradient of the reverse KL can be approximated as the difference between two score functions:

$$\begin{aligned}\nabla_\phi \mathcal{L}_{\text{DMD}} &\triangleq \mathbb{E}_t \left(\nabla_\phi \text{KL} \left(p_{\text{gen},t} \| p_{\text{data},t} \right) \right) \\ &\approx -\mathbb{E}_t \Bigg(\int \Big(s_{\text{data}} \left(\Psi \left(G_\phi(\epsilon), t \right), t \right) \\ &\qquad - s_{\text{gen}} \left(\Psi \left(G_\phi(\epsilon), t \right), t \right) \Big) \frac{dG_\phi(\epsilon)}{d\phi}\, d\epsilon \Bigg)\,,\end{aligned} \tag{4}$$

where Ψ represents the forward diffusion process as defined in Eq. 1, ϵ is random Gaussian noise, G_ϕ is the generator parameterized by ϕ, and s_{data} and s_{gen} represent the score functions trained on the data and generator's output distribution, respectively, using a denoising loss (Eq. 2).

During training, DMD [102] initializes both score functions from a pre-trained diffusion model. The score function of the data distribution is frozen, while the score function of the generator distribution is trained online using the generator's outputs. Simultaneously, the generator receives gradients to align its output with the data distribution (Eq. 4). DMD2 [101] extends this framework from single-step to multi-step generation by replacing the pure random noise input ϵ with a partially denoised intermediate image x_t.

4. Methods

Our approach builds upon an autoregressive diffusion transformer that enables causal generation (Sec. 4.1). We show our training procedure in Fig. 5, which uses asymmetric distillation (Sec. 4.2) and ODE initialization (Sec. 4.3) to achieve high generation quality and stable convergence. We achieve efficient streaming inference through KV caching mechanisms (Sec. 4.4).

4.1. Causal Architecture

We begin by compressing the video into a latent space using a 3D VAE. The VAE encoder processes each chunk of video frames independently, compressing them into shorter chunks of latent frames. The decoder then reconstructs the original video frames from each latent chunk. Our causal diffusion transformer operates in this latent space,

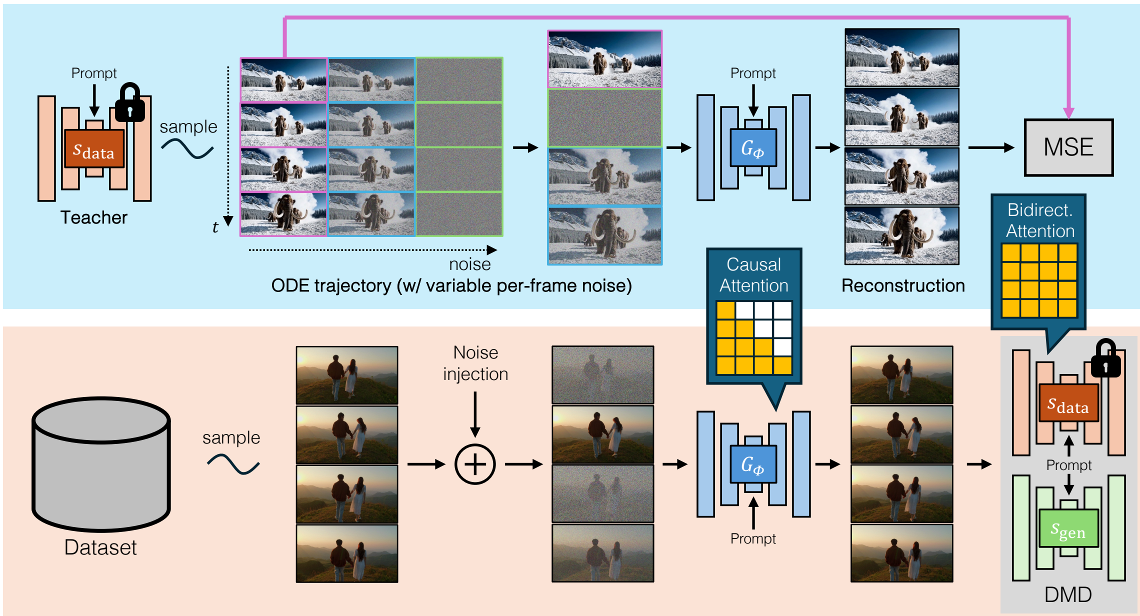


Figure 5. Our method distills a many-step, bidirectional video diffusion model s_{data} into a 4-step, causal generator G_ϕ. The training process consists of two stages. (top) Student Initialization: we initialize the causal student by pretraining it on a small set of ODE solution pairs generated by the bidirectional teacher (Sec. 4.3). This step helps stabilize the subsequent distillation training. (bottom) Asymmetric Distillation: using the *bidirectional* teacher, we train the *causal* student generator through a distribution matching distillation loss (Sec. 4.2).

generating latent frames sequentially. We design a block-wise causal attention mechanism inspired by prior works that combine autoregressive models with diffusion [39, 41, 50, 112]. Within each chunk, we apply bidirectional self-attention among latent frames to capture local temporal dependencies and maintain consistency. To enforce causality, we apply causal attention across chunks. This prevents frames in the current chunk from attending to frames in future chunks. Our design maintains the same latency as fully causal attention, as the VAE decoder still requires at least a block of latent frames to generate pixels. Formally, we define the attention mask M as

$$M_{i,j} = \begin{cases} 1, & \text{if } \left\lfloor \frac{j}{k} \right\rfloor \leq \left\lfloor \frac{i}{k} \right\rfloor, \\ 0, & \text{otherwise.} \end{cases} \tag{5}$$

Here, i and j index the frames in the sequence, k is the chunk size, and $\lfloor \cdot \rfloor$ denotes the floor function.

Our diffusion model G_ϕ extends the DiT architecture [61] for autoregressive video generation. We introduce block-wise causal attention masks to the self-attention layers (illustrated in Fig. 5) while preserving the core structure, allowing us to leverage pretrained bidirectional weights for faster convergence.

Algorithm 1 Asymmetric Distillation with DMD

Require: Few-step denoising timesteps $\mathcal{T} = \{0, t_1, t_2, \dots, t_Q\}$, video length N, chunk size k, pretrained bidirectional teacher model s_{data}, dataset $\mathcal{D}$.

```
Initialize student model G_φ with ODE regression (Sec. 4.3)
Initialize generator output's score function s_gen with s_data
while training do
    Sample a video from the dataset and divide frames into L = ⌈N/k⌉ chunks, {x_0^i}_{i=1}^L ~ D
    Sample per-chunk timesteps {t^i}_{i=1}^L ~ Uniform(T)
    Add noise: x_{t^i}^i = α_{t^i} x_0^i + σ_{t^i} ε^i, ε^i ~ N(0, I)
    Predict clean frames with the student: x̂_0^i = G_φ(x_{t^i}^i, t^i)
    Concatenate predictions: x̂_0 = [x̂_0^1, x̂_0^2, ..., x̂_0^L]
    Sample a single timestep t ~ Uniform(0, T)
    Add noise to predictions: x̂_t = α_t x̂_0 + σ_t ε, ε ~ N(0, I)
    Update student model G_φ using DMD loss L_DMD          ▷ Eq. 4
    Train generator's score function s_gen:
        Sample new noise ε' ~ N(0, I)
        Generate noisy x̂_t: x̂_t = α_t x̂_0 + σ_t ε'
        Compute denoising loss and update s_gen.           ▷ Eq. 2
end while
```

4.2. Bidirectional → Causal Generator Distillation

A straightforward approach to training a few-step causal generator would be through distillation from a causal teacher model. This involves adapting a pretrained bidirectional DiT model by incorporating causal attention mechanism described above and fine-tuning it using the denoising loss (Eq. 2). During training, the model takes as input a sequence of N noisy video frames divided into L chunks $\{x_t^i\}_{i=1}^L$, where $i \in \{1, 2, ..., L\}$ denotes the chunk index. Each chunk x_t^i has its own noise time step $t^i \sim [0, 999]$, following Diffusion Forcing [8]. During inference, the model denoises each chunk sequentially, conditioned on the previously generated clean chunks of frames. While distilling this fine-tuned autoregressive diffusion teacher appears promising in theory, our initial experiments indicated that this naive approach yields suboptimal results. Since causal diffusion models typically underperform their bidirectional counterparts, training a student model from a weaker causal teacher inherently limits the student's capabilities. Moreover, issues such as error accumulation would propagate from teacher to student. To overcome the limitations of a causal teacher, we propose an *asymmetric distillation* approach: following state-of-the-art video models [5, 63], we employ bidirectional attention in the teacher model while constraining the student model to causal attention (Fig. 5 bottom). Algorithm 1 details our training process.

4.3. Student Initialization

Directly training the causal student model using the DMD loss can be unstable due to architectural differences. To address this, we introduce an efficient initialization strategy to stabilize training (Fig. 5 top).

We create a small dataset of ODE solution pairs generated by the bidirectional teacher model:

- Sample a sequence of noise inputs $\{x_T^i\}_{i=1}^L$ from the standard Gaussian distribution $\mathcal{L}(0, I)$.
- Simulate the reverse diffusion process with an ordinary differential equation (ODE) solver [75] using the pre-trained bidirectional teacher model to obtain the corresponding ODE trajectory $\{x_t^i\}_{i=1}^L$, where t spans T to 0, covering all inference timesteps.

From the ODE trajectories, we select a subset of t values that match those used in our student generator. The student model is then trained on this dataset with a regression loss:

$$\mathcal{L}_{\text{init}} = \mathbb{E}_{x,t^i} ||G_\phi(\{x_{t^i}^i\}_{i=1}^N, \{t^i\}_{i=1}^N) - \{x_0^i\}_{i=1}^N||^2, \quad (6)$$

where the few-step generator G_ϕ is initialized from the teacher model. Our ODE initialization is computationally efficient, requiring only a small number of training iterations on relatively few ODE solution pairs.

4.4. Efficient Inference with KV Caching

During inference, we generate video frames sequentially using our autoregressive diffusion transformer with KV caching for efficient computation [6]. We show the detailed inference procedure in Algorithm 2. Notably, because we employ KV caching, block-wise causal attention is no longer needed at inference time. This allows us to leverage a fast bidirectional attention implementation [13].

Algorithm 2 Inference Procedure with KV Caching

Require: Denoising timesteps $\{t_0 = 0, t_1, \dots, t_Q\}$, video length N, chunk size k, few-step autoregressive video generator G_ϕ,
Divide frames into $L = \lceil N/k \rceil$ chunks
Initialize KV cache C $\leftarrow \emptyset$
for $i = 1$ to L **do**
  Initialize current chunk: $x_{t_Q}^i \sim \mathcal{N}(0, I)$
  Iterative denoising over timesteps:
  for $j = Q$ to 1 **do**
    Generate output: $\hat{x}_{t_j}^i = G_\phi(x_{t_j}^i, t_j)$ using cache C
    Update chunk: $x_{t_{j-1}}^i = \alpha_{t_{j-1}} \hat{x}_{t_j}^i + \sigma_{t_{j-1}} \epsilon'$
  end for
  Update KV cache:
  Compute KV pairs with a forward pass $G_\phi(x_0^i, 0)$
  Append new KV pairs to cache C
end for
Return $\{x_0^i\}_{i=1}^L$

5. Experiments

Models. Our teacher model is a bidirectional DiT [61] with an architecture similar to CogVideoX [98]. The model is trained on the latent space produced by a 3D VAE that encodes 16 video frames into a latent chunk consisting of 5 latent frames. The model is trained on 10-second videos at a resolution of 352×640 and 12 FPS. Our student model has the same architecture as the teacher, except that it employs causal attention where each token can only attend to other tokens within the same chunk and in preceding chunks. Each chunk contains 5 latent frames. During inference, it generates one chunk at a time using 4 denoising steps, with inference timesteps uniformly sampled as $[999, 748, 502, 247]$. We use FlexAttention [21] for efficient attention computation during training.

Training. We distill our causal student model using a mixed set of image and video datasets following CogVideoX [98]. Images and videos are filtered based on safety and aesthetic scores [73]. All videos are resized and cropped to the training resolution (352×640) and we use around 400K single-shot videos from an internal dataset, for which we have full copyright. During training, we first generate 1000 ODE pairs (Sec. 4.3) and train the student model for 3000 iterations with AdamW [51] optimizer and a learning rate of 5×10^{-6}. After that, we train with our asymmetric DMD loss (Sec. 4.2) with the AdamW optimizer and a learning rate of 2×10^{-6} for 6000 iterations. We use a guidance scale of 3.5 and adopt the two time-scale update rule from DMD2 [101] with a ratio of 5. The whole training process takes around 2 days on 64 H100 GPUs.

Evaluation. Our method is evaluated on VBench [27], a benchmark for video generation with 16 metrics designed to systematically assess motion quality and semantic alignment. For our main results, we use the first 128 prompts from MovieGen [63] to generate videos and evaluate model performance on three primary aspects from VBench competition's evaluation suite. A comprehensive evaluation using all 946 prompts from VBench, covering all 16 metrics, is provided in the supplementary material. The inference times are measured on a H100 GPU.

5.1. Text to Video Generation

We evaluate the ability of our approach to generate short videos (5 to 10 seconds) and compare it against state-of-the-art methods: CogVideoX [98], OpenSORA [111], Pyramid Flow [29], and MovieGen [63]. As shown in Tab. 1, our method outperforms all baselines across all three key metrics: temporal quality, frame quality, and text alignment. Our model achieves the highest temporal quality score of 94.7, indicating superior motion consistency and dynamic quality. In addition, our method shows notable improvements in frame quality and text alignment, scoring 64.4 and 30.1, respectively. In the supplementary material, we present our method's performance on the VBench-Long leaderboard, achieving a total score of 84.27 and securing first place among all officially evaluated video generation models.

We further evaluate our model's performance through a human preference study. We select the first 29 prompts from the MovieGenBench dataset and collect ratings from independent evaluators using the Prolific platform. For each pair of compared models and each prompt, we collect 3 ratings from different evaluators, resulting in a total of 87 ratings per model pair. The evaluators choose the better video between the two generated videos based on the visual quality and semantic alignment with the input prompt. The specific questions and interface are shown in Fig. 10. For reproducibility, we use a fixed random seed of zero for all videos. As illustrated in Fig. 6, our model consistently outperforms baseline methods such as MovieGen, CogVideoX, and Pyramid Flow. Notably, our distilled model maintains performance comparable to the bidirectional teacher while offering orders of magnitude faster inference, validating the effectiveness of our approach.

We also compare our method with prior works designed for long video generation: Gen-L-Video [84], FreeNoise [65], StreamingT2V [23], FIFO-Diffusion [34], and Pyramid Flow [29]. We use a sliding window inference strategy, taking the final frames of the previous 10-second segment as context for generating the next segment. The same strategy is also applied to generate long videos using Pyramid Flow. Tab. 2 shows that our method outperforms all baselines in terms of temporal quality and frame-wise quality and is competitive in text alignment. It also successfully prevents error accumulation. As shown in Fig. 7, our method maintains image quality over time, while most autoregressive baselines suffer from quality degradation [8, 23, 29].

Tab. 3 compares the efficiency of our method with competing approaches [29, 98] and our bidirectional teacher diffusion model. Our method achieves a 160× reduction in latency and a 16× improvement in throughput compared to the similarly scaled CogVideoX [98].

https://github.com/Vchitect/VBench/tree/master/competitions

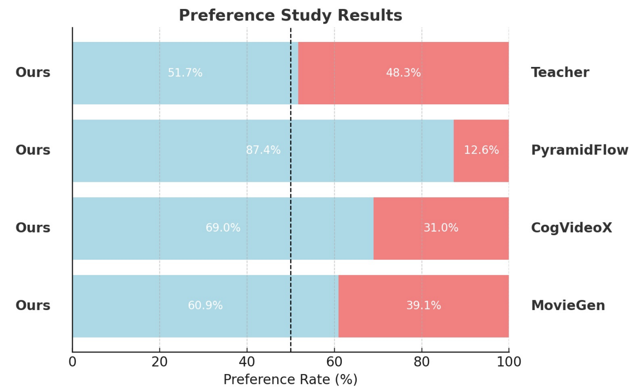


Figure 6. User study comparing our distilled causal video generator with its teacher model and existing video diffusion models. Our model demonstrates superior video quality (scores $> 50\%$), while achieving a significant reduction in latency by multiple orders of magnitude.

Method	Length (s)	Temporal Quality	Frame Quality	Text Alignment
CogVideoX-5B	6	89.9	59.8	29.1
OpenSORA	8	88.4	52.0	28.4
Pyramid Flow	10	89.6	55.9	27.1
MovieGen	10	91.5	61.1	28.8
CausVid (Ours)	10	**94.7**	**64.4**	**30.1**

Table 1. Evaluation of text-to-short-video generation. Each method is evaluated at its closest supported length to 10 seconds.

Method	Temporal Quality	Frame Quality	Text Alignment
Gen-L-Video	86.7	52.3	28.7
FreeNoise	86.2	54.8	28.7
StreamingT2V	89.2	46.1	27.2
FIFO-Diffusion	93.1	57.9	**29.9**
Pyramid Flow	89.0	48.3	24.4
CausVid (Ours)	**94.9**	**63.4**	28.9

Table 2. Evaluation of text-to-long-video generation. All methods produce videos approximately 30s in length.

5.2. Ablation Studies

First, we present results of directly fine-tuning the bidirectional DiT into a causal model, without using few-step distillation. We apply causal attention masks to the model

Method	Latency (s)	Throughput (FPS)
CogVideoX-5B	208.6	0.6
Pyramid Flow	6.7	2.5
Bidirectional Teacher	219.2	0.6
CausVid (Ours)	**1.3**	**9.4**

Table 3. Latency and throughput comparison across different methods for generating 10-second, 120-frame videos at a resolution of 640×352. The total time includes processing by the text encoder, diffusion model, and VAE decoder. Lower latency (↓) and higher throughput (↑) are preferred.

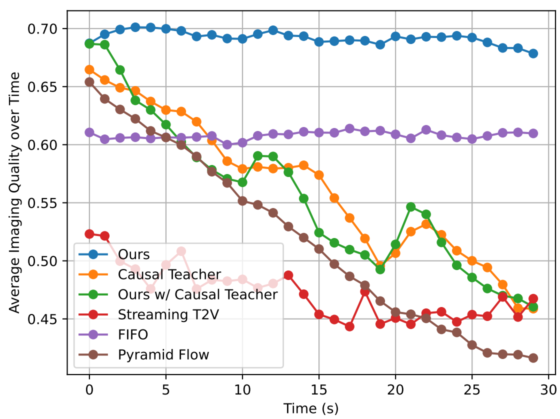


Figure 7. Imaging quality scores of generated videos over 30 seconds. Our distilled model and FIFO-Diffusion are the most effective at maintaining imaging quality over time. The sudden increase of score for the causal teacher around 20s is due to a switch of the sliding window, resulting in a temporary improvement in quality.

and fine-tune it with the autoregressive training method described in Sec. 4.2. As shown in Tab. 4, the many-step causal model performs substantially worse than the original bidirectional model. We observe that the causal baseline suffers from error accumulation, leading to rapid degradation in generation quality over time (orange in Fig. 7).

We then conduct an ablation study on our distillation framework, examining the student initialization scheme and the choice of teacher model. Tab. 4 shows that given the same ODE initialization scheme (as introduced in Sec. 4.3), the bidirectional teacher model outperforms the causal teacher model and is also much better than the initial ODE-fitted model (where we denote the teacher as None). As shown in Fig. 7, the causal diffusion teacher suffers from significant error accumulation (orange), which is then transferred to the student model (green). In contrast, we find that our causal student model trained with our asymmetric DMD loss and a bidirectional teacher (blue) performs much better than the many-step causal diffusion model, highlighting the importance of distillation for achieving both fast and high-quality video generation. With the same bidirectional teacher, we demonstrate that initializing the student model by fitting the ODE pairs can further enhance performance. While our student model improves upon the bidirectional teacher for frame-by-frame quality, it performs worse in temporal flickering and output diversity. Detailed discussions are included in the supplementary material.

Many-step models		Causal Generator?	# Fwd Pass	Temporal Quality	Frame Quality	Text Alignment
Bidirectional		✗	100	94.6	62.7	29.6
Causal		✓	100	92.4	60.1	28.5
Few-step models						
ODE Init.	Teacher					
✗	Bidirectional	✓	4	93.4	60.6	29.4
✓	None	✓	4	92.9	48.1	25.3
✓	Causal	✓	4	91.9	61.7	28.2
✓	Bidirectional	✓	4	**94.7**	**64.4**	**30.1**

Table 4. Ablation studies. All models generate videos of 10s. The top half presents results of fine-tuning the bidirectional DiT into causal models without few-step distillation. The bottom half compares different design choices in our distillation framework. The last row is our final configuration.

5.3. Applications

In addition to text-to-video generation, our method supports a broad range of other applications. We present quantitative results below, with qualitative samples in Fig. 2. We provide additional video results in the supplementary material.

Streaming Video-to-Video Translation. We evaluate our method on the task of streaming video-to-video translation, which aims to edit a streaming video input that can have unlimited frames. Inspired by SDEdit [59], we inject noise corresponding to timestep t_1 into each input video chunk and then denoise it in one step conditioned on the text. We compare our method with StreamV2V [42], a state-of-the-art method for this task that builds upon image diffusion models. From 67 video-prompt pairs used in StreamV2V's user study (originally from the DAVIS [64] dataset), we select all 60 videos that contain at least 16 frames. For a fair comparison, we do not apply any concept-specific fine-tuning to either method. Tab. 5 shows that our method outperforms StreamV2V, demonstrating improved temporal consistency due to the video prior in our model.

Method	Temporal Quality	Frame Quality	Text Alignment
StreamV2V	92.5	59.3	26.9
CausVid (Ours)	**93.2**	**61.7**	**27.7**

Table 5. Evaluation of streaming video-to-video translation.

Image to Video Generation Our model can perform text-conditioned image-to-video generation without any additional training. Given a text prompt and an initial image, we duplicate the image to create the first segment of frames. The model then autoregressively generates subsequent frames to extend the video. We achieve compelling results despite the simplicity of this approach. We evaluate against CogVideoX [98] and Pyramid Flow [29] on the VBench-I2V benchmark, as these are the primary baselines capable of generating 6-10 second videos. As shown in Table 6, our method outperforms existing approaches with notable improvements in dynamic quality. We believe instruction fine-tuning with a small set of image-to-video data could further enhance our model's performance.

Method	Temporal Quality	Frame Quality	Text Alignment
CogVideoX-5B	87.0	64.9	28.9
Pyramid Flow	88.4	60.3	27.6
CausVid (Ours)	**92.0**	**65.0**	**28.9**

Table 6. Evaluation of image-to-video generation. CogVideoX generates 6s video while the other methods generate 10s video.

6. Discussion

Although our method is able to generate high-quality videos up to 30 seconds, we still observe quality degradation when generating videos that are extremely long. Developing more effective strategies to address error accumulation remains future work. Moreover, while the latency is significantly lower—by multiple orders of magnitude—compared to previous approaches, it remains constrained by the current VAE design, which necessitates the generation of five latent frames before producing any output pixels. Adopting a more efficient frame-wise VAE could reduce latency by an additional order of magnitude, significantly improving the model's responsiveness. Finally, while our method produces high-quality samples using the DMD objective, it comes with reduced output diversity. This limitation is characteristic of reverse KL-based distribution matching approaches. Future work could explore alternative objectives such as EM-Distillation [94] and Score Implicit Matching [56], which may better preserve the diversity of outputs.

While our current implementation is limited to generating videos at around 10 FPS, standard engineering optimizations (including model compilation, quantization, and parallelization) could potentially enable real-time performance. We believe our work marks a significant advancement in video generation and opens up new possibilities for applications in robotic learning [14, 91], game rendering [7, 81], streaming video editing [9], and other scenarios that require real-time and long-horizon video generation.

Acknowledgment The research was partially supported by the Amazon Science Hub, GIST, Adobe, Google, as well as the United States Air Force Research Laboratory and the United States Air Force Artificial Intelligence Accelerator under Cooperative Agreement Number FA8750-19-2-1000. The views and conclusions expressed in this document are those of the authors and do not necessarily represent the official policies or endorsements of the United States Air Force or the U.S. Government. Notwithstanding any copyright statement herein, the U.S. Government is authorized to reproduce and distribute reprints for official purposes.

References

[1] Eloi Alonso, Adam Jelley, Vincent Micheli, Anssi Kanervisto, Amos Storkey, Tim Pearce, and François Fleuret. Diffusion for world modeling: Visual details matter in atari. In *NeurIPS*, 2024. 5

[2] Fan Bao, Chongxuan Li, Yue Cao, and Jun Zhu. All are worth words: a vit backbone for score-based diffusion models. In *NeurIPS 2022 Workshop on Score-Based Methods*, 2022. 1

[3] 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. 1

[4] 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. In *CVPR*, 2023. 6

[5] Tim Brooks, Bill Peebles, Connor Holmes, Will DePue, Yufei Guo, Li Jing, David Schnurr, Joe Taylor, Troy Luhman, Eric Luhman, Clarence Ng, Ricky Wang, and Aditya Ramesh. Video generation models as world simulators. 2024. 1, 5, 6, 8

[6] Tom B Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. In *NeurIPS*, 2020. 2, 5, 8

[7] Haoxuan Che, Xuanhua He, Quande Liu, Cheng Jin, and Hao Chen. Gamegen-x: Interactive open-world game video generation. *arXiv preprint arXiv:2411.00769*, 2024. 2, 11

[8] Boyuan Chen, Diego Marti Monso, Yilun Du, Max Simchowitz, Russ Tedrake, and Vincent Sitzmann. Diffusion forcing: Next-token prediction meets full-sequence diffusion. *arXiv preprint arXiv:2407.01392*, 2024. 4, 5, 8, 9

[9] Feng Chen, Zhen Yang, Bohan Zhuang, and Qi Wu. Streaming video diffusion: Online video editing with diffusion models. *arXiv preprint arXiv:2405.19726*, 2024. 11

[10] Haoxin Chen, Menghan Xia, Yingqing He, Yong Zhang, Xiaodong Cun, Shaoshu Yang, Jinbo Xing, Yaofang Liu, Qifeng Chen, Xintao Wang, et al. Videocrafter1: Open diffusion models for high-quality video generation. *arXiv preprint arXiv:2310.19512*, 2023. 6

[11] Shouyuan Chen, Sherman Wong, Liangjian Chen, and Yuandong Tian. Extending context window of large language models via positional interpolation. *arXiv preprint arXiv:2306.15595*, 2023. 16

[12] Xinyuan Chen, Yaohui Wang, Lingjun Zhang, Shaobin Zhuang, Xin Ma, Jiashuo Yu, Yali Wang, Dahua Lin, Yu Qiao, and Ziwei Liu. Seine: Short-to-long video diffusion model for generative transition and prediction. In *ICLR*, 2023. 5

[13] Tri Dao, Dan Fu, Stefano Ermon, Atri Rudra, and Christopher Ré. Flashattention: Fast and memory-efficient exact attention with io-awareness. In *NeurIPS*, 2022. 8

[14] Alejandro Escontrela, Ademi Adeniji, Wilson Yan, Ajay Jain, Xue Bin Peng, Ken Goldberg, Youngwoon Lee, Danijar Hafner, and Pieter Abbeel. Video prediction models as rewards for reinforcement learning. *NeurIPS*, 2024. 11

[15] 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 *ICML*, 2024. 5

[16] Jean-Yves Franceschi, Mike Gartrell, Ludovic Dos Santos, Thibaut Issenhuth, Emmanuel de Bézenac, Mickaël Chen, and Alain Rakotomamonjy. Unifying gans and score-based diffusion as generative particle models. In *NeurIPS*, 2023. 5

[17] Kaifeng Gao, Jiaxin Shi, Hanwang Zhang, Chunping Wang, and Jun Xiao. Vid-gpt: Introducing gpt-style autoregressive generation in video diffusion models. *arXiv preprint arXiv:2406.10981*, 2024. 5

[18] Songwei Ge, Thomas Hayes, Harry Yang, Xi Yin, Guan Pang, David Jacobs, Jia-Bin Huang, and Devi Parikh. Long video generation with time-agnostic vqgan and time-sensitive transformer. In *ECCV*, 2022. 5

[19] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks. *Communications of the ACM*, 2020. 5

[20] Zekun Hao, Xun Huang, and Serge Belongie. Controllable video generation with sparse trajectories. In *CVPR*, 2018. 5

[21] Horace He, Driss Guessous, Yanbo Liang, and Joy Dong. Flexattention: The flexibility of pytorch with the performance of flashattention. *PyTorch Blog*, 2024. 8

[22] Jonathan Heek, Emiel Hoogeboom, and Tim Salimans. Multistep consistency models. *arXiv preprint arXiv:2403.06807*, 2024. 5

[23] Roberto Henschel, Levon Khachatryan, Daniil Hayrapetyan, Hayk Poghosyan, Vahram Tadevosyan, Zhangyang Wang, Shant Navasardyan, and Humphrey Shi. Streamingt2v: Consistent, dynamic, and extendable long video generation from text. *arXiv preprint arXiv:2403.14773*, 2024. 5, 9

[24] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *NeurIPS*, 2020. 5, 6

[25] Jonathan Ho, Tim Salimans, Alexey Gritsenko, William Chan, Mohammad Norouzi, and David J Fleet. Video diffusion models. In *NeurIPS*, 2022. 1, 6

[26] Wenyi Hong, Ming Ding, Wendi Zheng, Xinghan Liu, and Jie Tang. Cogvideo: Large-scale pretraining for text-to-video generation via transformers. In *ICLR*, 2023. 6

[27] Ziqi Huang, Yinan He, Jiashuo Yu, Fan Zhang, Chenyang Si, Yuming Jiang, Yuanhan Zhang, Tianxing Wu, Qingyang Jin, Nattapol Chanpaisit, et al. Vbench: Comprehensive benchmark suite for video generative models. In *CVPR*, 2024. 8, 16

[28] Allan Jabri, David Fleet, and Ting Chen. Scalable adaptive computation for iterative generation. In *ICML*, 2023. 6

[29] Yang Jin, Zhicheng Sun, Ningyuan Li, Kun Xu, Hao Jiang, Nan Zhuang, Quzhe Huang, Yang Song, Yadong Mu, and Zhouchen Lin. Pyramidal flow matching for efficient video generative modeling. *arXiv preprint arXiv:2410.05954*, 2024. 1, 2, 4, 5, 9, 11

[30] Minguk Kang, Richard Zhang, Connelly Barnes, Sylvain Paris, Suha Kwak, Jaesik Park, Eli Shechtman, Jun-Yan Zhu, and Taesung Park. Distilling diffusion models into conditional gans. In *ECCV*, 2024. 5

[31] Tero Karras, Miika Aittala, Timo Aila, and Samuli Laine. Elucidating the design space of diffusion-based generative models. In *NeurIPS*, 2022. 6

[32] Tero Karras, Miika Aittala, Jaakko Lehtinen, Janne Hellsten, Timo Aila, and Samuli Laine. Analyzing and improving the training dynamics of diffusion models. In *CVPR*, 2024. 6

[33] Dongjun Kim, Chieh-Hsin Lai, Wei-Hsiang Liao, Naoki Murata, Yuhta Takida, Toshimitsu Uesaka, Yutong He, Yuki Mitsufuji, and Stefano Ermon. Consistency trajectory models: Learning probability flow ode trajectory of diffusion. In *ICLR*, 2024. 5

[34] Jihwan Kim, Junoh Kang, Jinyoung Choi, and Bohyung Han. Fifo-diffusion: Generating infinite videos from text without training. *arXiv preprint arXiv:2405.11473*, 2024. 5, 9

[35] Diederik Kingma, Tim Salimans, Ben Poole, and Jonathan Ho. Variational diffusion models. In *NeurIPS*, 2021. 6

[36] Akio Kodaira, Chenfeng Xu, Toshiki Hazama, Takanori Yoshimoto, Kohei Ohno, Shogo Mitsuhori, Soichi Sugano, Hanying Cho, Zhijian Liu, and Kurt Keutzer. Streamdiffusion: A pipeline-level solution for real-time interactive generation. *arXiv preprint arXiv:2312.12491*, 2023. 5

[37] Dan Kondratyuk, Lijun Yu, Xiuye Gu, José Lezama, Jonathan Huang, Grant Schindler, Rachel Hornung, Vighnesh Birodkar, Jimmy Yan, Ming-Chang Chiu, et al. Videopoet: A large language model for zero-shot video generation. 2024. 2, 5

[38] Alex X Lee, Richard Zhang, Frederik Ebert, Pieter Abbeel, Chelsea Finn, and Sergey Levine. Stochastic adversarial video prediction. *arXiv preprint arXiv:1804.01523*, 2018. 5

[39] Sangyun Lee, Gayoung Lee, Hyunsu Kim, Junho Kim, and Youngjung Uh. Diffusion models with grouped latents for interpretable latent space. In *ICML Workshop*, 2023. 7

[40] Jiachen Li, Weixi Feng, Tsu-Jui Fu, Xinyi Wang, Sugato Basu, Wenhu Chen, and William Yang Wang. T2v-turbo: Breaking the quality bottleneck of video consistency model with mixed reward feedback. In *NeurIPS*, 2024. 5

[41] Tianhong Li, Yonglong Tian, He Li, Mingyang Deng, and Kaiming He. Autoregressive image generation without vector quantization. In *NeurIPS*, 2024. 7

[42] Feng Liang, Akio Kodaira, Chenfeng Xu, Masayoshi Tomizuka, Kurt Keutzer, and Diana Marculescu. Looking backward: Streaming video-to-video translation with feature banks. *arXiv preprint arXiv:2405.15757*, 2024. 5, 10

[43] Jian Liang, Chenfei Wu, Xiaowei Hu, Zhe Gan, Jianfeng Wang, Lijuan Wang, Zicheng Liu, Yuejian Fang, and Nan Duan. Nuwa-infinity: Autoregressive over autoregressive generation for infinite visual synthesis. In *NeurIPS*, 2022. 5

[44] Shanchuan Lin and Xiao Yang. Animatediff-lightning: Cross-model diffusion distillation. *arXiv preprint arXiv:2403.12706*, 2024. 5

[45] Shanchuan Lin, Anran Wang, and Xiao Yang. Sdxl-lightning: Progressive adversarial diffusion distillation. *arXiv*, 2024. 5

[46] Haozhe Liu, Wentian Zhang, Jinheng Xie, Francesco Faccio, Mengmeng Xu, Tao Xiang, Mike Zheng Shou, Juan-Manuel Perez-Rua, and Jürgen Schmidhuber. Faster diffusion via temporal attention decomposition. *arXiv e-prints*, pages arXiv–2404, 2024. 6

[47] Xingchao Liu, Chengyue Gong, and Qiang Liu. Flow straight and fast: Learning to generate and transfer data with rectified flow. In *ICLR*, 2023. 5

[48] Xingchao Liu, Xiwen Zhang, Jianzhu Ma, Jian Peng, and Qiang Liu. Instaflow: One step is enough for high-quality diffusion-based text-to-image generation. In *ICLR*, 2024. 5

[49] Ziwei Liu, Raymond A Yeh, Xiaoou Tang, Yiming Liu, and Aseem Agarwala. Video frame synthesis using deep voxel flow. In *CVPR*, 2017. 5

[50] Zhijun Liu, Shuai Wang, Sho Inoue, Qibing Bai, and Haizhou Li. Autoregressive diffusion transformer for text-to-speech synthesis. *arXiv preprint arXiv:2406.05551*, 2024. 7

[51] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *ICLR*, 2019. 8

[52] Cheng Lu, Yuhao Zhou, Fan Bao, Jianfei Chen, Chongxuan Li, and Jun Zhu. Dpm-solver: A fast ode solver for diffusion probabilistic model sampling in around 10 steps. In *NeurIPS*, 2022. 6

[53] Eric Luhman and Troy Luhman. Knowledge distillation in iterative generative models for improved sampling speed. *arXiv preprint arXiv:2101.02388*, 2021. 5

[54] Simian Luo, Yiqin Tan, Longbo Huang, Jian Li, and Hang Zhao. Latent consistency models: Synthesizing high-resolution images with few-step inference. *arXiv preprint arXiv:2310.04378*, 2023. 5

[55] Weijian Luo, Tianyang Hu, Shifeng Zhang, Jiacheng Sun, Zhenguo Li, and Zhihua Zhang. Diff-instruct: A universal approach for transferring knowledge from pre-trained diffusion models. In *NeurIPS*, 2023. 5

[56] Weijian Luo, Zemin Huang, Zhengyang Geng, J Zico Kolter, and Guo-jun Qi. One-step diffusion distillation through score implicit matching. *arXiv preprint arXiv:2410.16794*, 2024. 11

[57] Xiaofeng Mao, Zhengkai Jiang, Fu-Yun Wang, Wenbing Zhu, Jiangning Zhang, Hao Chen, Mingmin Chi, and Yabiao Wang. Osv: One step is enough for high-quality image to video generation. *arXiv preprint arXiv:2409.11367*, 2024. 5

[58] Michael Mathieu, Camille Couprie, and Yann LeCun. Deep multi-scale video prediction beyond mean square error. In *ICLR*, 2016. 5

[59] Chenlin Meng, Yutong He, Yang Song, Jiaming Song, Jiajun Wu, Jun-Yan Zhu, and Stefano Ermon. Sdedit: Guided image synthesis and editing with stochastic differential equations. In *ICLR*, 2021. 10

[60] Chenlin Meng, Robin Rombach, Ruiqi Gao, Diederik Kingma, Stefano Ermon, Jonathan Ho, and Tim Salimans. On distillation of guided diffusion models. In *CVPR*, 2023. 5

[61] William Peebles and Saining Xie. Scalable diffusion models with transformers. In *ICCV*, 2023. 1, 6, 7, 8

[62] Bowen Peng, Jeffrey Quesnelle, Honglu Fan, and Enrico Shippole. Yarn: Efficient context window extension of large language models. *arXiv preprint arXiv:2309.00071*, 2023. 16

[63] Adam Polyak, Amit Zohar, Andrew Brown, Andros Tjandra, Animesh Sinha, Ann Lee, Apoorv Vyas, Bowen Shi, Chih-Yao Ma, Ching-Yao Chuang, et al. Movie gen: A cast of media foundation models. *arXiv preprint arXiv:2410.13720*, 2024. 1, 5, 8, 9

[64] Jordi Pont-Tuset, Federico Perazzi, Sergi Caelles, Pablo Arbeláez, Alex Sorkine-Hornung, and Luc Van Gool. The 2017 davis challenge on video object segmentation. *arXiv preprint arXiv:1704.00675*, 2017. 10

[65] Haonan Qiu, Menghan Xia, Yong Zhang, Yingqing He, Xintao Wang, Ying Shan, and Ziwei Liu. Freenoise: Tuning-free longer video diffusion via noise rescheduling. In *ICLR*, 2024. 5, 9

[66] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, 2019. 4

[67] Yuxi Ren, Xin Xia, Yanzuo Lu, Jiacheng Zhang, Jie Wu, Pan Xie, Xing Wang, and Xuefeng Xiao. Hyper-sd: Trajectory segmented consistency model for efficient image synthesis. 2024. 5

[68] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *CVPR*, 2022. 6

[69] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *MICCAI*, 2015. 6

[70] David Ruhe, Jonathan Heek, Tim Salimans, and Emiel Hoogeboom. Rolling diffusion models. In *ICML*, 2024. 5

[71] Tim Salimans and Jonathan Ho. Progressive distillation for fast sampling of diffusion models. In *ICLR*, 2022. 5, 6

[72] Axel Sauer, Dominik Lorenz, Andreas Blattmann, and Robin Rombach. Adversarial diffusion distillation. In *ECCV*, 2024. 5

[73] Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, et al. Laion-5b: An open large-scale dataset for training next generation image-text models. In *NeurIPS*, 2022. 8

[74] Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *ICML*, 2015. 6

[75] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. In *ICLR*, 2021. 5, 6, 8

[76] Yang Song and Prafulla Dhariwal. Improved techniques for training consistency models. In *ICLR*, 2024. 5

[77] Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. In *ICLR*, 2021. 6

[78] Yang Song, Prafulla Dhariwal, Mark Chen, and Ilya Sutskever. Consistency models. In *ICML*, 2023. 5

[79] Zhenxiong Tan, Xingyi Yang, Songhua Liu, and Xinchao Wang. Video-infinity: Distributed long video generation. *arXiv preprint arXiv:2406.16260*, 2024. 5

[80] Sergey Tulyakov, Ming-Yu Liu, Xiaodong Yang, and Jan Kautz. Mocogan: Decomposing motion and content for video generation. In *CVPR*, pages 1526–1535, 2018. 5

[81] Dani Valevski, Yaniv Leviathan, Moab Arar, and Shlomi Fruchter. Diffusion models are real-time game engines. *arXiv preprint arXiv:2408.14837*, 2024. 5, 11

[82] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NeurIPS*, 2017. 6

[83] Carl Vondrick and Antonio Torralba. Generating the future with adversarial transformers. In *CVPR*, 2017. 5

[84] Fu-Yun Wang, Wenshuo Chen, Guanglu Song, Han-Jia Ye, Yu Liu, and Hongsheng Li. Gen-l-video: Multi-text to long video generation via temporal co-denoising. *arXiv preprint arXiv:2305.18264*, 2023. 5, 9

[85] Fu-Yun Wang, Zhaoyang Huang, Qiang Ma, Xudong Lu, Bianwei Kang, Yijin Li, Yu Liu, and Hongsheng Li. Zola: Zero-shot creative long animation generation with short video model. In *ECCV*, 2024. 5

[86] Fu-Yun Wang, Zhaoyang Huang, Xiaoyu Shi, Weikang Bian, Guanglu Song, Yu Liu, and Hongsheng Li. Animatelcm: Accelerating the animation of personalized diffusion models and adapters with decoupled consistency learning. *arXiv preprint arXiv:2402.00769*, 2024. 5

[87] Xiang Wang, Shiwei Zhang, Han Zhang, Yu Liu, Yingya Zhang, Changxin Gao, and Nong Sang. Videolcm: Video latent consistency model. *arXiv preprint arXiv:2312.09109*, 2023. 5

[88] Yuqing Wang, Tianwei Xiong, Daquan Zhou, Zhijie Lin, Yang Zhao, Bingyi Kang, Jiashi Feng, and Xihui Liu. Loong: Generating minute-level long videos with autoregressive language models. *arXiv preprint arXiv:2410.02757*, 2024. 5

[89] Zhengyi Wang, Cheng Lu, Yikai Wang, Fan Bao, Chongxuan Li, Hang Su, and Jun Zhu. Prolificdreamer: High-fidelity and diverse text-to-3d generation with variational score distillation. In *NeurIPS*, 2023. 5

[90] Wenming Weng, Ruoyu Feng, Yanhui Wang, Qi Dai, Chunyu Wang, Dacheng Yin, Zhiyuan Zhao, Kai Qiu, Jianmin Bao, Yuhui Yuan, et al. Art-v: Auto-regressive text-to-video generation with diffusion models. In *CVPRW*, 2024. 5

[91] Hongtao Wu, Ya Jing, Chilam Cheang, Guangzeng Chen, Jiafeng Xu, Xinghang Li, Minghuan Liu, Hang Li, and Tao Kong. Unleashing large-scale video generative pre-training for visual robot manipulation. In *ICLR*, 2024. 11

[92] Yecheng Wu, Zhuoyang Zhang, Junyu Chen, Haotian Tang, Dacheng Li, Yunhao Fang, Ligeng Zhu, Enze Xie, Hongxu Yin, Li Yi, et al. Vila-u: a unified foundation model integrating visual understanding and generation. *arXiv preprint arXiv:2409.04429*, 2024. 5

[93] Desai Xie, Zhan Xu, Yicong Hong, Hao Tan, Difan Liu, Feng Liu, Arie Kaufman, and Yang Zhou. Progressive autoregressive video diffusion models. *arXiv preprint arXiv:2410.08151*, 2024. 5

[94] Sirui Xie, Zhisheng Xiao, Diederik P Kingma, Tingbo Hou, Ying Nian Wu, Kevin Patrick Murphy, Tim Salimans, Ben Poole, and Ruiqi Gao. Em distillation for one-step diffusion models. *arXiv preprint arXiv:2405.16852*, 2024. 11

[95] Zhening Xing, Gereon Fox, Yanhong Zeng, Xingang Pan, Mohamed Elgharib, Christian Theobalt, and Kai Chen. Live2diff: Live stream translation via uni-directional attention in video diffusion models. *arXiv preprint arXiv:2407.08701*, 2024. 5

[96] Yanwu Xu, Yang Zhao, Zhisheng Xiao, and Tingbo Hou. Ufogen: You forward once large scale text-to-image generation via diffusion gans. In *CVPR*, 2024. 5

[97] Wilson Yan, Yunzhi Zhang, Pieter Abbeel, and Aravind Srinivas. Videogpt: Video generation using vq-vae and transformers. *arXiv preprint arXiv:2104.10157*, 2021. 5

[98] 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, 5, 6, 8, 9, 11

[99] Mingxuan Yi, Zhanxing Zhu, and Song Liu. Monoflow: Rethinking divergence gans via the perspective of wasserstein gradient flows. In *ICML*, 2023. 5

[100] Shengming Yin, Chenfei Wu, Huan Yang, Jianfeng Wang, Xiaodong Wang, Minheng Ni, Zhengyuan Yang, Linjie Li, Shuguang Liu, Fan Yang, et al. Nuwa-xl: Diffusion over diffusion for extremely long video generation. In *ACL*, 2023. 5

[101] Tianwei Yin, Michaël Gharbi, Taesung Park, Richard Zhang, Eli Shechtman, Fredo Durand, and William T Freeman. Improved distribution matching distillation for fast image synthesis. In *NeurIPS*, 2024. 4, 5, 6, 8

[102] Tianwei Yin, Michaël Gharbi, Richard Zhang, Eli Shechtman, Frédo Durand, William T Freeman, and Taesung Park. One-step diffusion with distribution matching distillation. In *CVPR*, 2024. 4, 5, 6

[103] Qinsheng Zhang and Yongxin Chen. Fast sampling of diffusion models with exponential integrator. In *ICLR*, 2023. 6

[104] Qinsheng Zhang, Jiaming Song, Xun Huang, Yongxin Chen, and Ming-Yu Liu. Diffcollage: Parallel generation of large content with diffusion models. In *CVPR*. IEEE, 2023. 5

[105] Wentian Zhang, Haozhe Liu, Jinheng Xie, Francesco Faccio, Mike Zheng Shou, and Jürgen Schmidhuber. Cross-attention makes inference cumbersome in text-to-image diffusion models. *arXiv preprint arXiv:2404.02747*, 2024. 6

[106] Zhicheng Zhang, Junyao Hu, Wentao Cheng, Danda Paudel, and Jufeng Yang. Extdm: Distribution extrapolation diffusion model for video prediction. In *CVPR*, 2024. 5

[107] Zhixing Zhang, Yanyu Li, Yushu Wu, Yanwu Xu, Anil Kag, Ivan Skorokhodov, Willi Menapace, Aliaksandr Siarohin, Junli Cao, Dimitris Metaxas, et al. Sf-v: Single forward video generation model. *arXiv preprint arXiv:2406.04324*, 2024. 5

[108] Zhixing Zhang, Bichen Wu, Xiaoyan Wang, Yaqiao Luo, Luxin Zhang, Yinan Zhao, Peter Vajda, Dimitris Metaxas, and Licheng Yu. Avid: Any-length video inpainting with diffusion model. In *CVPR*, 2024. 5

[109] Canyu Zhao, Mingyu Liu, Wen Wang, Jianlong Yuan, Hao Chen, Bo Zhang, and Chunhua Shen. Moviedreamer: Hierarchical generation for coherent long visual sequence. *arXiv preprint arXiv:2407.16655*, 2024. 5

[110] Xuanlei Zhao, Xiaolong Jin, Kai Wang, and Yang You. Real-time video generation with pyramid attention broadcast. *arXiv preprint arXiv:2408.12588*, 2024. 6

[111] 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, 2024. 1, 5, 6, 9

[112] Chunting Zhou, Lili Yu, Arun Babu, Kushal Tirumala, Michihiro Yasunaga, Leonid Shamis, Jacob Kahn, Xuezhe Ma, Luke Zettlemoyer, and Omer Levy. Transfusion: Predict the next token and diffuse images with one multi-modal model. *arXiv preprint arXiv:2408.11039*, 2024. 7

[113] Daquan Zhou, Weimin Wang, Hanshu Yan, Weiwei Lv, Yizhe Zhu, and Jiashi Feng. Magicvideo: Efficient video generation with latent diffusion models. *arXiv preprint arXiv:2211.11018*, 2022. 6

[114] Chang Zou, Xuyang Liu, Ting Liu, Siteng Huang, and Linfeng Zhang. Accelerating diffusion transformers with token-wise feature caching. *arXiv preprint arXiv:2410.05317*, 2024. 6

From Slow Bidirectional to Fast Autoregressive Video Diffusion Models

Supplementary Material

A. VBench-Long Leaderboard Results

We evaluate CausVid on the VBench-Long dataset using all 946 prompts across 16 standardized metrics. We refer readers to the VBench paper [27] for a detailed description of the metrics. As shown in Tab. 7, our method achieves state-of-the-art performance with the highest total score of 84.27. The radar plot in Fig. 8 visualizes our method's comprehensive performance advantages. Our method is significantly ahead in several key metrics including dynamic degree, aesthetic quality, imaging quality, object class, multiple objects, and human action. More details can be found on the official benchmark website (`https://huggingface.co/spaces/Vchitect/VBench_Leaderboard`).

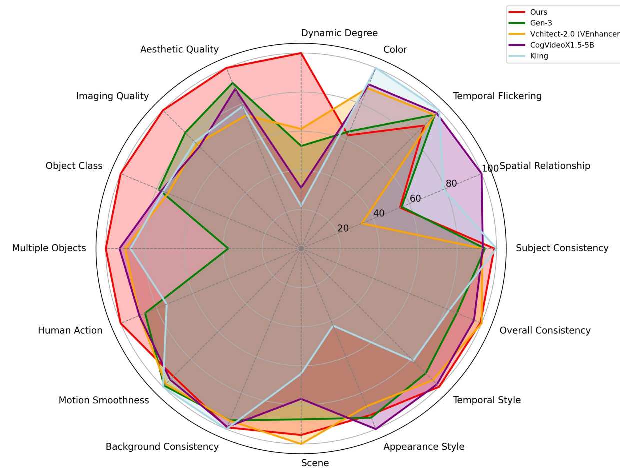
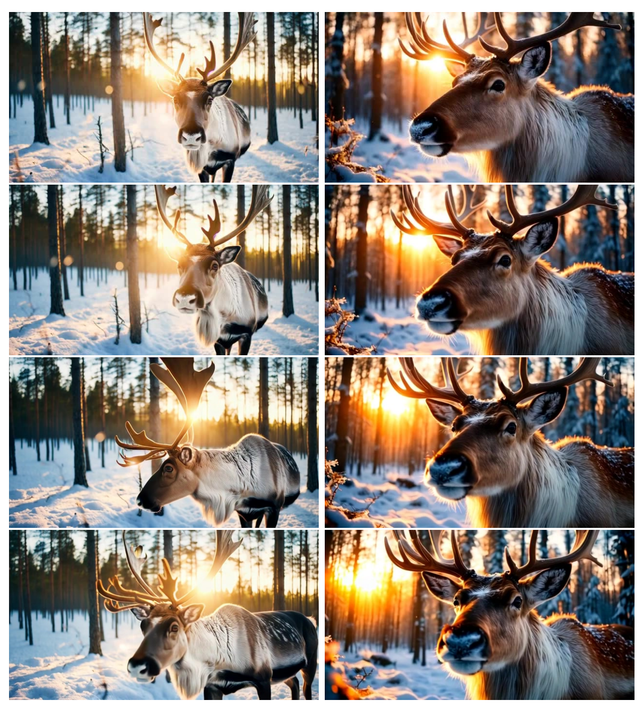


Figure 8. VBench metrics and comparison with previous state-of-the-art methods. Our method (red) performs strongly across different dimensions.

B. Qualitative Comparison with the Teacher

As demonstrated by VBench (Tab. 4) and human evaluations (Fig. 6), our distilled causal model obtains comparable overall quality to the bidirectional diffusion teacher model. In Fig. 9, we show qualitative comparisons between the two models. Additional qualitative results can be found on our project website (`https://causvid.github.io/`).



Bidirectional Teacher Ours

Figure 9. Qualitative comparison with the teacher.

C. Limitations

C.1. Long-range Inconsistency

While our method demonstrates high-quality long video generation (see supplementary website), it faces a key limitation: the sliding window inference strategy discards context frames beyond a 10-second horizon. This temporal truncation can cause visual inconsistencies when previously-seen objects or environments reappear after extended periods. A potential solution could involve supervised fine-tuning on a curated set of long videos, inspired by practices in large language model training [11, 62].

Method	Total Score	Quality Score	Semantic Score	Subject Consistency	Background Consistency	Temporal Flickering	Motion Smoothness	Dynamic Degree	Aesthetic Quality	Imaging Quality	Object Class	Multiple Objects	Human Action	Color	Spatial Relationship	Scene	Appearance Style	Temporal Style	Overall Consistency
Vchitect	82.24	83.54	77.06	96.83	96.66	98.57	98.98	63.89	60.41	65.35	86.61	68.84	97.20	87.04	57.55	**56.57**	23.73	25.01	27.57
Jimeng	81.97	83.29	76.69	97.25	98.39	99.03	98.09	38.43	**68.80**	67.09	89.62	69.08	90.10	89.05	77.45	44.94	22.27	24.7	27.10
CogVideoX	81.61	82.75	77.04	96.23	96.52	98.66	96.92	70.97	61.98	62.90	85.23	62.11	99.40	82.81	66.35	53.20	**24.91**	**25.38**	**27.59**
Vidu	81.89	83.85	74.04	94.63	96.55	99.08	97.71	82.64	60.87	63.32	88.43	61.68	97.40	83.24	66.18	46.07	21.54	23.79	26.47
Kling	81.85	83.39	75.68	**98.33**	97.60	**99.30**	**99.40**	46.94	61.21	65.62	87.24	68.05	93.40	**89.90**	73.03	50.86	19.62	24.17	26.42
CogVideoX1.5-5B	82.17	82.78	**79.76**	96.87	**97.35**	98.88	98.31	50.93	62.79	65.02	87.47	69.65	97.20	87.55	**80.25**	52.91	24.89	25.19	27.30
Gen-3	82.32	84.11	75.17	97.10	96.62	98.61	99.23	60.14	63.34	66.82	87.81	53.64	96.40	80.90	65.09	54.57	24.31	24.71	26.69
CausVid (Ours)	**84.27**	**85.65**	78.75	97.53	97.19	96.24	98.05	**92.69**	64.15	**68.88**	**92.99**	**72.15**	**99.80**	80.17	64.65	56.58	24.27	25.33	27.51

Table 7. Full comparison on VBench-Long using all 16 metrics. The best scores are in bold. Please zoom in for details.

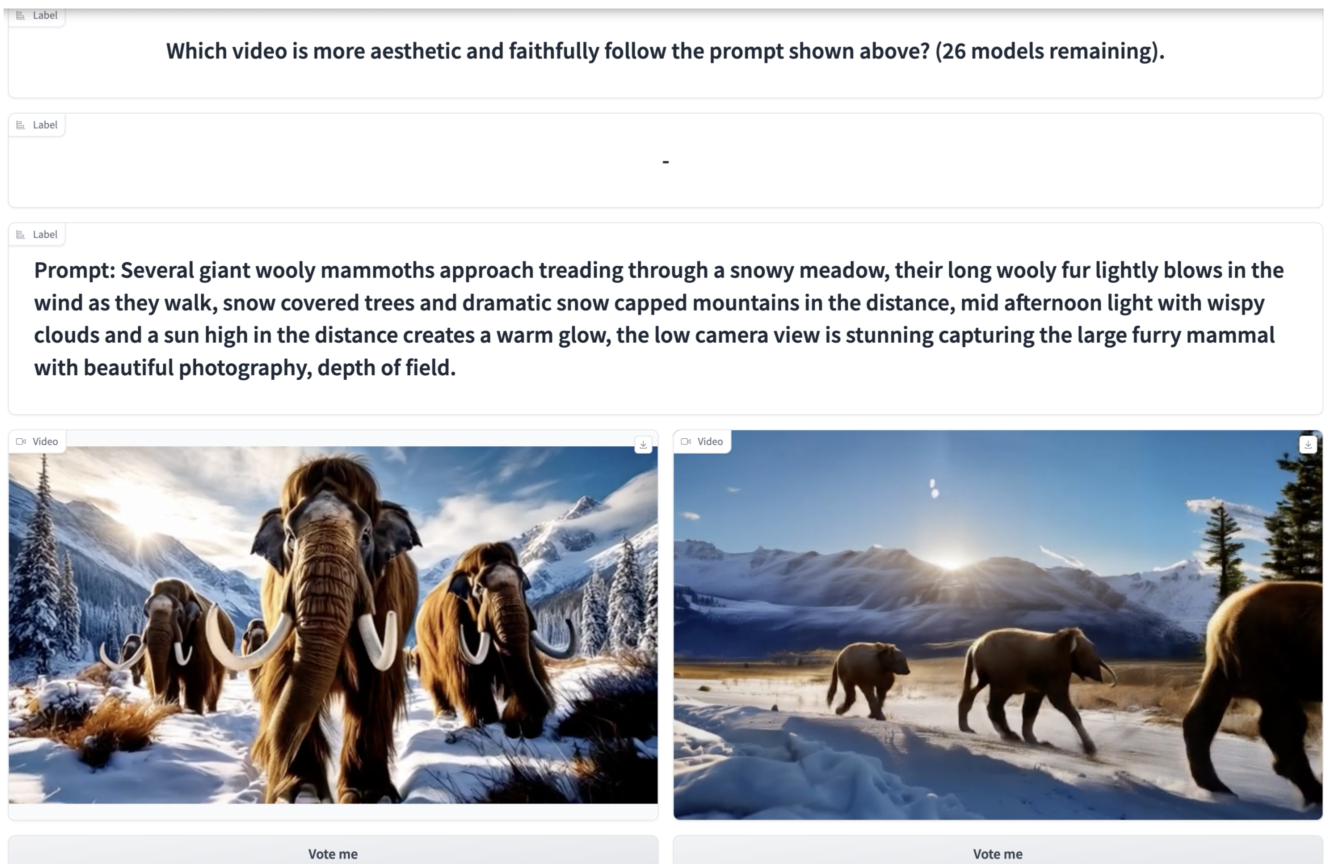


Figure 10. An example interface for our user preference study, where videos generated by different methods are displayed in a randomized left/right arrangement.