

# MASK FORCING: IMPROVING AUTOREGRESSIVE VIDEO DIFFUSION DISTILLATION VIA DUAL-NOISE MASKING ROLLOUT

Zhuoran Zhao<sup>1,2</sup>, Shengju Qian<sup>3</sup>, Tongtong Liang<sup>4</sup>, Xianghao Kong<sup>2</sup>, Songchun Zhang<sup>2</sup>, Junchao Huang<sup>5</sup>, Guian Fang<sup>6</sup>, Xin Wang<sup>3</sup>, Pan Hui<sup>1,2</sup>, Anyi Rao<sup>2</sup>

<sup>1</sup>HKUST(GZ) <sup>2</sup>HKUST <sup>3</sup>LIGHTSPEED <sup>4</sup>UCSD <sup>5</sup>CUHK(SZ) <sup>6</sup>NUS

## ABSTRACT

Autoregressive (AR) video diffusion models have shown great potential in real-time video generation. Recent methods distill pretrained bidirectional video diffusion models into causal AR students through Distribution Matching Distillation (DMD), but the generated videos often suffer from over-saturation and over-smoothing issues, resulting in limited visual quality and realism. The key contributing factor is the mode-seeking behavior of the reverse KL objective in DMD, which can cause the student distribution to collapse onto only a few modes of the teacher distribution. To address this, we propose Mask Forcing, a Dual-Noise Masking Rollout strategy that perturbs the AR student self-rollout to mitigate mode collapse induced by reverse-KL mode seeking. The core idea is to inject cleaner signals into noisy rollout inputs via random masks along spatial and temporal axes during the self-rollout process of AR diffusion distillation. Such perturbations encourage the student rollouts to explore more regions of the teacher distribution, allowing DMD to provide learning signals beyond the modes already covered by the student. Moreover, the cleaner tokens act as denoising guidance for other noisier tokens, improving the intermediate rollout predictions and reducing error accumulation. Extensive experiments demonstrate that our method improves multiple AR video diffusion distillation methods with higher visual quality efficiently, without incorporating real video data or additional post-training stages. Project page: <https://alicezrzhaio.github.io/mask-forcing>.

## 1 INTRODUCTION

Video diffusion models have advanced rapidly in recent years, enabling the generation of long-duration and high-fidelity videos (Wan et al., 2025; Kong et al., 2024; HaCohen et al., 2026). However, these models typically rely on bidirectional attention and multiple denoising timesteps, limiting their applicability to real-time streaming generation scenarios. Therefore, autoregressive (AR) diffusion models have emerged as a promising paradigm for real-time video generation (Huang et al., 2026b; Zhu et al., 2026; Yang et al., 2025; Hu et al., 2026; Gao et al., 2026). By leveraging causal attention mechanisms, AR diffusion models can generate future chunks sequentially, conditioning each chunk on previously generated ones via causal dependencies. This formulation supports real-time inference and unbounded video generation without recomputing previously generated frames.

Teacher Forcing (Hu et al., 2024; Gao et al., 2025) and Diffusion Forcing (Chen et al., 2024) are two representative paradigms for training causal autoregressive video diffusion models. Teacher Forcing trains the model to predict the next frames conditioned on the ground-truth frames, leading to exposure bias since the model can only condition on its own predictions during inference. Diffusion Forcing instead trains the model by assigning each frame with independently sampled noise. While this paradigm alleviates the distribution shift, it still fails to align training with inference. Self Forcing (Huang et al., 2026b) bridges the train-test gap by training the model with self-rollout, generating the next frame based on previously self-generated frames rather than ground-truth context and distilling the teacher’s knowledge via a DMD loss (Yin et al., 2024b). However, a critical

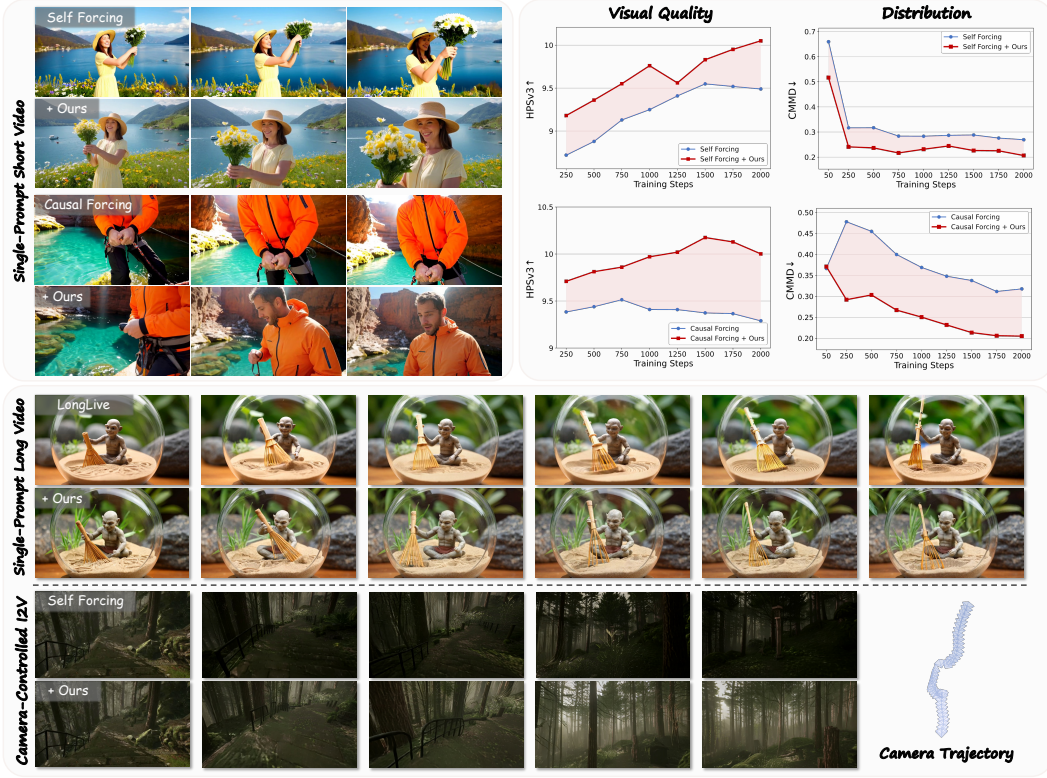


Figure 1: Mask Forcing improves autoregressive video generation through a Dual-Noise Masking Rollout strategy that alleviates the mode-seeking behavior of the reverse KL objective in DMD, enhancing video visual quality efficiently without incorporating real video data or additional post-training. We show consistent improvement in visual realism across short-video, long-video, and camera-controlled autoregressive generation.

limitation of combining self-rollout with a DMD loss is that it often produces over-saturated and over-smoothed videos, exhibiting low visual quality and limited realism.

This phenomenon can be attributed to two key factors. First, the reverse-KL objective in DMD exhibits mode-seeking behavior that tends to cover only the high-probability regions of the teacher’s distribution (Chen et al., 2025; Cai et al., 2026; Zheng et al., 2026b). This behavior induces mode collapse, leading to reduced diversity in the generated videos. Second, the intermediate rollout predictions receive no explicit training signal at each step and suffer from error accumulation. Recent methods mitigate these issues by explicitly incorporating real data into the training objective (Yin et al., 2024a; Chen et al., 2026; Liu et al., 2026), but they rely on complex data-curation processes or additional post-training, which further complicates the multi-stage training pipeline. Another line of research incorporates reinforcement learning post-training to improve the quality of the distilled AR model (Zhang et al., 2026), but its performance still relies on reward models and is sensitive to multiple hyperparameters. A complementary line of work balances mode-seeking and mode-covering objectives (Cai et al., 2026; Zheng et al., 2026b; Li et al., 2026a), but the generated videos can still appear over-saturated and lack fine-grained visual details. This begs the question: can we improve AR video diffusion distillation without real data curation or additional post-training stages?

To address this question, we focus on the self-rollout trajectory, which determines both the student samples exposed to DMD and the intermediate predictions reused in subsequent denoising and autoregressive conditioning. Perturbing the rollout trajectory can expose DMD to a broader range of student samples, allowing it to provide learning signals beyond the modes the student already covers. Moreover, cleaner tokens can serve as context for denoising noisier tokens (Chefer et al., 2026), helping improve intermediate rollout predictions and reduce error accumulation. Based on these motivations, we propose Mask Forcing, which injects randomly masked cleaner tokens into the roll-



out input to mitigate both mode-seeking behavior and error accumulation. At each rollout transition, Mask Forcing randomly samples a mask and an additional timestep corresponding to a lower noise level than the original one and injects the cleaner tokens into the rollout input. Specifically, the latent positions selected by the mask are re-noised to this lower noise level, while the remaining positions are kept at the original level, producing a dual-noise rollout input. The conditioning timestep fed to the model remains the original one, requiring the model to denoise inputs whose local noise levels are partially inconsistent with the global timestep. Randomizing both the mask and the lower-noise timestep diversifies the student rollout trajectory and broadens the region of sample space explored during training, enabling DMD to provide learning signals from teacher modes not yet covered by the student. Beyond perturbing the rollout trajectory, Mask Forcing uses lower-noise tokens as context for denoising noisier tokens. This design is consistent with the observation in Self-Flow (Chefer et al., 2026) that cleaner context in mixed-noise inputs can facilitate denoising, thereby improving intermediate predictions and reducing error accumulation during self-rollout.

Extensive experiments across multiple AR video distillation methods validate the effectiveness of our method in both chunk-wise and frame-wise settings. Representative results in Fig. 1 show that our method significantly improves the visual quality against various baselines, with enhanced realism and richer high-frequency details. Comprehensive evaluations demonstrate improvements across multiple benchmarks. In summary, our contributions are as follows:

- We propose Mask Forcing, a simple and effective approach that alleviates the mode-seeking behavior of the reverse-KL objective in self-rollout DMD training for AR video diffusion distillation, without incorporating real video data or additional post-training stages.
- We introduce a Dual-Noise Masking Rollout strategy that injects lower-noise signals into noisy rollout inputs through random masks along spatial and temporal axes. Such perturbations diversify student rollout trajectories to cover more teacher modes, while providing cleaner context for denoising to reduce error accumulation.
- Extensive experiments on multiple AR video distillation methods demonstrate the effectiveness of Mask Forcing in both chunk-wise and frame-wise settings, with significantly improved visual quality and faster convergence. Comprehensive ablations further validate the effects of different masking mechanisms.

## 2 RELATED WORK

### 2.1 AUTOREGRESSIVE VIDEO GENERATION

Autoregressive video diffusion models enable real-time video generation by sequentially generating videos conditioned on historical context. Teacher Forcing (Hu et al., 2024; Gao et al., 2025) denoises the current chunk conditioned on clean ground-truth context, which suffers from a train-test gap and exposure bias. Diffusion Forcing (Chen et al., 2024) assigns each frame with independently sampled noise to approximate rollout distributions, but still fails to align training with inference. Self Forcing (Huang et al., 2026b) bridges this gap by performing AR self-rollout on self-generated histories and distills a bidirectional teacher into the causal student via a DMD loss. Causal Forcing (Zhu et al., 2026) further uses an AR teacher for ODE initialization to reduce the architecture gap. LongLive (Yang et al., 2025) extends causal AR generation to long videos via short window attention with frame sink and streaming long tuning. Despite these advances, distilled AR models still exhibit limited visual quality and realism, motivating methods that introduce additional training signals, real data, or post-training stages. DMD2 (Yin et al., 2024a) introduces a GAN loss and real training data, but suffers from training instability and additional real data curation. DFD (Chen et al., 2026) integrates real data into the distillation score, but requires post-training on a DMD2-pretrained model. Astrolabe (Zhang et al., 2026) explores RL post-training on distilled AR models, but performance is limited by reward models. In contrast, Mask Forcing mitigates mode collapse by perturbing student rollouts, improving AR distillation without real video supervision or post-training.

### 2.2 MODE SEEKING AND MODE COVERING IN VIDEO DIFFUSION DISTILLATION

DMD (Yin et al., 2024b) and DMD2 (Yin et al., 2024a) use score-based reverse KL distribution matching to align student-generated samples with the teacher distribution. This mode-seeking ob-

jective can improve sample fidelity but may concentrate the student distribution on a limited subset of teacher modes. DMD2 further introduces an adversarial loss on real data. In contrast, trajectory-based consistency objectives are commonly associated with the mode-covering behavior of forward divergence (Song et al., 2023; Kim et al., 2024; Lu & Song, 2025). Such objectives can cover more teacher modes, but may average across modes and exhibit lower sample quality. Recent methods balance these behaviors by combining complementary objectives. Mode Seeking meets Mean Seeking (Cai et al., 2026) uses separate heads for supervised flow matching on long videos and reverse-KL distribution matching against a short-video teacher. rCM (Zheng et al., 2026b) augments continuous-time consistency with score distillation regularization. DistillAlign (Li et al., 2026a) analyzes the initialization effects and jointly optimizes DMD and a consistency distillation loss. Unlike these methods, Mask Forcing retains the original DMD objective and instead perturbs the student rollouts via dual-noise masking. This broadens the student distribution exposed to DMD, allowing it to provide learning signals from teacher modes not reached by standard self-rollout.

### 2.3 MASKED MODELING

Masked modeling has become a powerful paradigm for representation and generative learning in computer vision (He et al., 2022; Bao et al., 2021; Chang et al., 2022). The core idea is to mask a portion of the input and train the model to recover it. In representation learning, MAE (He et al., 2022) masks a large portion of random patches from the input image and reconstructs them in pixel space by leveraging context from visible parts. In generative modeling, MaskGIT (Chang et al., 2022) adopts a mask-then-predict objective with parallel iterative decoding to synthesize images. Masking can also be realized through heterogeneous noise levels that control the information retained by each token. For multi-modal generation, Self-Flow (Chefer et al., 2026) introduces a self-supervised framework for flow matching that combines mix-timestep scheduling with masking for representation alignment. In AR generation, Diffusion Forcing associates each frame with a random, independent noise level, which can be regarded as partial masking along the time axis. Inspired by these works, we introduce dual-noise masking into AR video distillation to diversify self-rollout trajectories and provide cleaner context for denoising noisier tokens.

## 3 METHOD

### 3.1 PRELIMINARIES

**Autoregressive (AR) Video Generation.** An AR video model represents a video as a sequence of  $F$  chunks  $x_{1:F} = (x_1, \dots, x_F)$ , where each chunk may include one or more latent frames. It factorizes the text-conditioned joint distribution as  $p(x_{1:F} | c) = \prod_{i=1}^F p(x_i | x_{<i}, c)$ , where  $c$  denotes the text prompt and  $x_{<i} = (x_1, \dots, x_{i-1})$  denotes the preceding video chunks. Each conditional distribution  $p(x_i | x_{<i}, c)$  is modeled using a diffusion process following the flow-matching formulation, where the noisy sample is defined as  $x_i^t = (1-t)x_i + t\epsilon_i$ , with  $\epsilon_i \sim \mathcal{N}(0, I)$  and  $t \in [0, 1]$ . Each video chunk is generated through this denoising process conditioned on  $c$  and the historical context  $x_{<i}$ , which is stored in the key-value (KV) cache. Teacher Forcing (TF) and Diffusion Forcing (DF) are two typical training paradigms for AR video models using frame-wise MSE loss between predicted and ground-truth targets. In TF, the timestep  $t$  is shared across all frames, and the context consists of clean ground-truth frames. In DF, each frame is assigned an independently sampled timestep  $t_i$ , and the historical context is noisy. To mitigate the train-test gap and alleviate error accumulation, Self Forcing unrolls the model on its own generated samples during training, with  $x_{1:F}^\theta \sim \prod_{i=1}^F p_\theta(x_i | x_{<i}, c)$ .

**Distribution Matching Distillation (DMD).** DMD (Yin et al., 2024b;a) distills a pretrained multi-step teacher model into a few-step student model  $G_\theta$  by minimizing the reverse KL divergence from the student generator induced distribution  $p_{\text{fake}}$  to the teacher distribution  $p_{\text{real}}$ . Specifically, DMD adopts a reverse KL objective, whose gradient is used to update the student model:

$$\nabla_\theta \mathcal{L}_{\text{DMD}} = \mathbb{E}[\nabla_\theta D_{KL}(p_{\text{fake},t} || p_{\text{real},t})], \quad (1)$$

$$D_{KL}(p_{\text{fake},t} || p_{\text{real},t}) = \mathbb{E} \left[ \log \frac{p_{\text{fake},t}(x_t)}{p_{\text{real},t}(x_t)} \right] = -\mathbb{E}[\log p_{\text{real},t}(x_t) - \log p_{\text{fake},t}(x_t)], \quad (2)$$

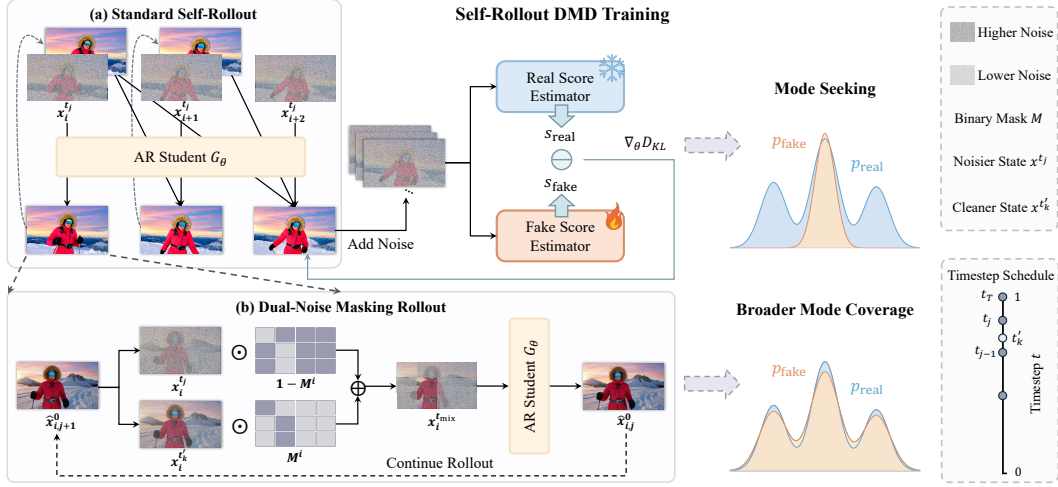


Figure 2: (a) Standard self-rollout DMD training suffers from mode collapse due to the mode-seeking reverse-KL objective, resulting in over-saturation and limited realism. (b) Mask Forcing perturbs the student rollout trajectory through dual-noise masking rollout, encouraging broader teacher-mode coverage while providing cleaner context for denoising noisy tokens.

where  $x_t$  is a noisy sample corresponding to timestep  $t$ :  $x_t = (1 - t)G_\theta(z) + t\epsilon$  with  $\epsilon \sim \mathcal{N}(0, I)$ , and  $z$  is latent drawn from random noise. The gradient to update  $G$  is formulated as the difference between two score functions:

$$\nabla_\theta \mathcal{L}_{\text{DMD}} = \mathbb{E} \left[ -(s_{\text{real}}(x_t, t) - s_{\text{fake}}(x_t, t)) \frac{dG}{d\theta} \right], \quad (3)$$

where  $s_{\text{real}}(x_t, t) = \nabla_{x_t} \log p_{\text{real},t}(x_t)$ ,  $s_{\text{fake}}(x_t, t) = \nabla_{x_t} \log p_{\text{fake},t}(x_t)$  denote the scores of the teacher and student distributions, respectively. However, the reverse KL objective is inherently mode-seeking and may concentrate the student distribution on a limited set of teacher modes, leading to over-saturation and reduced realism (Chen et al., 2026).

### 3.2 DUAL-NOISE MASKING ROLLOUT

#### 3.2.1 OVERVIEW

We propose a dual-noise masking rollout strategy for the self-rollout DMD training to mitigate mode collapse induced by the mode-seeking behavior of reverse KL in AR diffusion distillation. The core idea is to inject low-noise signals into noisy rollout inputs during the rollout process via random masks applied both within and across chunks. These cleaner signals serve two purposes. First, they perturb the student rollouts, encouraging broader coverage of high-density regions in the teacher distribution and thereby mitigating mode collapse and visual artifacts such as over-saturation and over-smoothing. Second, cleaner tokens provide context for denoising noisier tokens, following the principle of masked modeling. An overview of our method is shown in Fig. 2.

#### 3.2.2 SELF-ROLLOUT WITH DUAL-NOISE MASKING

An AR video diffusion model represents a video as a sequence of  $F$  chunks  $x = (x_1, x_2, \dots, x_F)$ , where each chunk contains one or more latent frames. Using the flow-matching formulation, a noisy chunk is defined as:

$$x_i^t = (1 - t)x_i^0 + t\epsilon, \quad \epsilon \sim \mathcal{N}(0, I), \quad (4)$$

where  $t \in [0, 1]$  denotes the timestep, which interpolates between the clean chunk at  $t = 0$  and pure Gaussian noise at  $t = 1$ . During the self-rollout training, the student model  $G_\theta$  generates the video chunk by chunk and each chunk is produced by iterative denoising over a fixed schedule of  $T$  timesteps  $\{t_1, \dots, t_T\}$  selected from the  $N_t = 1000$  training timesteps, where  $t_T = 1$  denotes pure noise,  $t_0 = 0$  denotes the clean output. We use  $T = 4$  denoising steps throughout training. For each



chunk index  $i$ , at denoising timestep  $t_j$ , the student model predicts a clean chunk estimate  $\hat{x}_{i,j}^0$  from the noisy chunk  $x_i^{t_j}$ , conditioned on the previously generated clean chunks  $x_{<i}$ , the text prompt  $c$ , and the timestep  $t_j$ :

$$\hat{x}_{i,j}^0 = G_\theta \left( x_i^{t_j} \mid x_{<i}, c, t_j \right). \quad (5)$$

The DMD objective is evaluated only on the completed self-rollout, providing no explicit training signal for intermediate predictions at each step. Since intermediate predictions are reused in subsequent denoising steps and as context for later chunks, their errors can accumulate throughout the rollout. Meanwhile, the reverse-KL objective is inherently mode-seeking and concentrates the student rollout distribution on a narrow set of high-density modes of the teacher distribution. Together, these limitations may contribute to degraded visual quality and realism in generated videos.

We address this issue by injecting cleaner signals into the student model input to perturb the student rollout trajectory, encouraging the student rollouts to reach more high-density regions of the teacher distribution. Specifically, we adopt a dual-timestep scheduling strategy inspired by Self-Flow (Chefer et al., 2026). We retain the denoising schedule of the base model and define  $\Delta$  as the size of the timestep window measured in training timestep units, corresponding to a normalized width of  $\Delta/N_t$ . At each denoising timestep  $t_j$ , we uniformly sample an additional timestep  $t'_k$  within the corresponding window in timestep space:

$$\ell_j = \max \left( t_{\min}, t_j - \frac{\Delta}{N_t} \right), \quad (6)$$

$$t'_k \sim \text{Uniform}([\ell_j, t_j]). \quad (7)$$

Here,  $t'_k$  corresponds to a lower noise level than  $t_j$ , and the floor  $t_{\min}$  prevents the injected signal from being nearly clean.

For each chunk  $i$ , we construct a binary mask  $M^i$  with a masking ratio  $\alpha$ . The mask is sampled independently across frames within a chunk (spatial axis) and across chunks (temporal axis), so that different frames of the same chunk and different chunks receive different dual-noise patterns. We observe that such mask diversity affects the visual quality and motion dynamics balance of the generated videos. At the  $j$ -th denoising step, corresponding to timestep  $t_j$ , we sample  $\epsilon_j \sim \mathcal{N}(0, I)$  and use it to re-noise the clean prediction from the previous step  $\hat{x}_{i,j+1}^0$  at  $t'_k$  and  $t_j$ , yielding the cleaner sample  $x_i^{t'_k}$  and the noisier sample  $x_i^{t_j}$ , respectively:

$$x_i^{t'_k} = (1 - t'_k) \hat{x}_{i,j+1}^0 + t'_k \epsilon_j, \quad (8)$$

$$x_i^{t_j} = (1 - t_j) \hat{x}_{i,j+1}^0 + t_j \epsilon_j. \quad (9)$$

We use the random mask  $M^i$  to select tokens from the cleaner sample at masked positions ( $M^i = 1$ ), while retaining tokens from the original noisier sample at the remaining positions. This yields the dual-noise input:

$$x_i^{t_{\text{mix}}} = M^i \odot x_i^{t'_k} + (1 - M^i) \odot x_i^{t_j}. \quad (10)$$

Consequently, the student generator  $G_\theta$  receives the dual-noise input at denoising step  $j$  and produces the updated clean prediction:

$$\hat{x}_{i,j}^0 = G_\theta \left( x_i^{t_{\text{mix}}} \mid x_{<i}, c, t_j \right). \quad (11)$$

The student generator is conditioned on the scheduled timestep  $t_j$ , while cleaner tokens in the dual-noise input are treated as a training perturbation. The model follows the original denoising schedule at inference. At the next denoising step,  $\hat{x}_{i,j}^0$  is re-noised at the scheduled timestep  $t_{j-1}$  and a newly sampled cleaner timestep within its corresponding timestep window. The two noisy samples are then combined using the mask  $M^i$  to form the next dual-noise input, replacing the standard re-noising operation in self-rollout. Once the current chunk reaches the sampled exit step  $s$ , the corresponding clean prediction  $\hat{x}_{i,s}^0$  is used to update the KV cache and condition subsequent chunks.

This dual-noise perturbation achieves two goals. First, it diversifies the student rollout trajectories, so that the distillation score in the DMD loss is evaluated over a broader region of the sample space rather than a narrow subset of teacher modes. This encourages the student model to explore more diverse modes and prevents it from collapsing onto the high-density modes of the teacher model induced by the reverse KL objective. Second, since the masked tokens carry lower noise, the student model is guided to exploit them as context, helping denoise the noisier tokens and improving intermediate rollout predictions.

### 3.2.3 DISTRIBUTIONAL ANALYSIS

We analyze the student rollout distribution induced by dual-noise masking. We characterize it as a mixture over masking trajectories and decompose its reverse-KL objective using mutual information.

Fix a text condition  $c$  and a DMD score-noising timestep  $\tau$ , distinct from the denoising timesteps used during self-rollout. Let  $X_\tau$  denote the completed student rollout noised at  $\tau$  and let  $p_\tau$  denote the corresponding teacher noisy marginal. The masking trajectory  $V$  is defined as the collection of all masks and lower-noise timesteps sampled during the rollout, with  $V \sim \pi$ , where  $\pi$  is determined by the mask ratio, timestep window, and mask sampling scheme.

Conditioned on a masking trajectory  $V = v$ ,  $q_{\theta,\tau}^v(x)$  denotes the corresponding student distribution. Marginalizing over  $V$  gives the masked-rollout distribution  $\bar{q}_{\theta,\tau}(x) := \mathbb{E}_{V \sim \pi}[q_{\theta,\tau}^V(x)]$ . The dependence between  $V$  and  $X_\tau$  conditioned on  $c$  is measured by their mutual information, given by

$$I_\theta(V; X_\tau | c) = \mathbb{E}_{V \sim \pi} [D_{\text{KL}}(q_{\theta,\tau}^V \| \bar{q}_{\theta,\tau})]. \quad (12)$$

The reverse-KL objective of the masked-rollout distribution can be decomposed as

$$D_{\text{KL}}(\bar{q}_{\theta,\tau} \| p_\tau) = \mathbb{E}_{V \sim \pi} [D_{\text{KL}}(q_{\theta,\tau}^V \| p_\tau)] - I_\theta(V; X_\tau | c). \quad (13)$$

$D_{\text{KL}}(\bar{q}_{\theta,\tau} \| p_\tau)$  is the divergence between the marginal masked-rollout distribution and the teacher marginal.  $\mathbb{E}_{V \sim \pi}[D_{\text{KL}}(q_{\theta,\tau}^V \| p_\tau)]$  is the average divergence between each trajectory-conditioned distribution and the teacher marginal, while  $I_\theta(V; X_\tau | c)$  measures their output diversity.

When different masking trajectories induce distinct conditional distributions,  $I_\theta(V; X_\tau | c) > 0$ . Individual trajectory-conditioned distributions may remain locally concentrated on different teacher-supported regions, while their marginal mixture can collectively cover these regions. This explains how marginalizing over trajectories can alleviate mode-seeking behavior in the overall student distribution without requiring individual conditional distributions to be mode-covering. The detailed derivation is provided in Appendix 6.1.

The mask ratio, timestep window, and mask sampling scheme determine  $\pi$  and the strength of the induced rollout perturbations. Under weak perturbations, different masking trajectories induce similar output distributions, resulting in small  $I_\theta(V; X_\tau | c)$ , and limited additional teacher-mode coverage. Stronger perturbations can produce more distinct output distributions and increase  $I_\theta(V; X_\tau | c)$ , but excessive perturbations can move them away from teacher-supported regions and increase their average reverse KL. These factors balance trajectory diversity and teacher alignment. We study their effects through ablations in Sec. 4.2.

## 4 EXPERIMENTS

**Implementation Details.** We adopt Self Forcing (Huang et al., 2026b), LongLive (Yang et al., 2025), and Causal Forcing (Zhu et al., 2026) as our baselines, using Wan2.1-T2V-1.3B (Wan et al., 2025) as the base model and Wan2.1-T2V-14B as the teacher. Our method is applied during the self-rollout DMD training stage. The generated videos consist of 81 frames at a resolution of  $832 \times 480$ , and the training prompts are sampled from the VidProM dataset (Wang & Yang, 2024). We implement each baseline under both chunk-wise and frame-wise autoregressive generation settings, with 3 latent frames per chunk in the chunk-wise setting. Since Self Forcing and LongLive do not release their frame-wise ODE initialization models, we train these models using ODE-paired data distilled from the bidirectional teacher. For Causal Forcing, we use its causal student model initialized via ODE distillation from the autoregressive teacher. During training, we set the mask ratio  $\alpha$  to 0.2, the timestep window  $\Delta$  to 250, and the floor  $t_{\min}$  to the value at schedule index 20. In the chunk-wise setting, the mask is sampled independently across frames within a chunk and across chunks, while in the frame-wise setting, the mask is sampled independently across chunks. This training procedure takes approximately 1.5k steps and 14 hours on 8 GPUs.

**Evaluation.** We adopt the 100-prompt set with rich motion and complex actions from Causal Forcing and VBench as our primary evaluation benchmarks. For the 100-prompt set, we use HPSv3 (Ma et al., 2025) to evaluate the overall visual quality and employ VisionReward (Xu et al., 2026) to report vision reward score (Vision.), instruction following (Instruct.), and motion quality (MQ) sub-scores following Causal Forcing. Dynamic Degree (Dynamic.) is used to evaluate motions in video



Figure 3: Qualitative comparison of our method (+Ours) against different baselines. Visual results show that our method generates videos with higher visual quality and greater realism, exhibiting fewer over-saturation artifacts and richer high-frequency details.

Table 1: Quantitative results on the 100-prompt set and VBench benchmarks under chunk-wise and frame-wise settings.

| Method            | HPSv3 $\uparrow$       | Vision. $\uparrow$     | Instruct. $\uparrow$   | MQ $\uparrow$          | Dynamic. $\uparrow$ | Total $\uparrow$       | Quality $\uparrow$     | Semantic $\uparrow$    |
|-------------------|------------------------|------------------------|------------------------|------------------------|---------------------|------------------------|------------------------|------------------------|
| <i>Chunk-wise</i> |                        |                        |                        |                        |                     |                        |                        |                        |
| Self Forcing      | 9.55                   | 10.10                  | 38.50                  | 15.88                  | 70                  | 81.89                  | 82.99                  | 77.49                  |
| + Ours            | 9.84 <sub>+0.29</sub>  | 11.37 <sub>+1.27</sub> | 45.03 <sub>+6.53</sub> | 20.49 <sub>+4.61</sub> | 82 <sub>+12</sub>   | 82.61 <sub>+0.72</sub> | 83.68 <sub>+0.69</sub> | 78.74 <sub>+1.25</sub> |
| Causal Forcing    | 9.37                   | 10.36                  | 40.41                  | 17.73                  | 76                  | 82.67                  | 83.58                  | 78.98                  |
| + Ours            | 10.17 <sub>+0.80</sub> | 11.58 <sub>+1.22</sub> | 46.30 <sub>+5.89</sub> | 21.54 <sub>+3.81</sub> | 82 <sub>+6</sub>    | 82.76 <sub>+0.09</sub> | 83.69 <sub>+0.11</sub> | 79.01 <sub>+0.03</sub> |
| LongLive          | 9.11                   | 10.77                  | 42.48                  | 21.20                  | 76                  | 82.02                  | 82.87                  | 78.66                  |
| + Ours            | 10.14 <sub>+1.03</sub> | 11.00 <sub>+0.23</sub> | 42.65 <sub>+0.17</sub> | 22.34 <sub>+1.14</sub> | 69 <sub>-7</sub>    | 82.75 <sub>+0.73</sub> | 83.71 <sub>+0.84</sub> | 78.91 <sub>+0.25</sub> |
| <i>Frame-wise</i> |                        |                        |                        |                        |                     |                        |                        |                        |
| Self Forcing      | 9.34                   | 9.45                   | 35.62                  | 18.27                  | 53                  | 80.73                  | 81.72                  | 76.78                  |
| + Ours            | 9.79 <sub>+0.45</sub>  | 10.49 <sub>+1.04</sub> | 39.60 <sub>+3.98</sub> | 19.07 <sub>+0.80</sub> | 61 <sub>+8</sub>    | 81.49 <sub>+0.76</sub> | 82.55 <sub>+0.83</sub> | 77.23 <sub>+0.45</sub> |
| Causal Forcing    | 9.67                   | 10.58                  | 37.56                  | 20.25                  | 28                  | 80.64                  | 81.55                  | 77.00                  |
| + Ours            | 9.96 <sub>+0.29</sub>  | 10.71 <sub>+0.13</sub> | 39.60 <sub>+2.04</sub> | 23.69 <sub>+3.44</sub> | 52 <sub>+24</sub>   | 82.28 <sub>+1.64</sub> | 83.19 <sub>+1.64</sub> | 78.62 <sub>+1.62</sub> |
| LongLive          | 9.19                   | 9.35                   | 38.50                  | 12.14                  | 25                  | 80.97                  | 81.86                  | 77.41                  |
| + Ours            | 9.46 <sub>+0.27</sub>  | 10.55 <sub>+1.20</sub> | 42.78 <sub>+4.28</sub> | 19.32 <sub>+7.18</sub> | 76 <sub>+51</sub>   | 81.47 <sub>+0.50</sub> | 82.33 <sub>+0.47</sub> | 78.05 <sub>+0.64</sub> |

with RAFT (Teed & Deng, 2020). All metrics are scaled by 100 for readability except for HPSv3. For VBench, we report the Total Score, Quality Score, and Semantic Score with the 946 standard prompts. For the long-video generation setting, we additionally evaluate our method against LongLive with the first 100 prompts from the MovieGen (Polyak et al., 2024) extended version and the VBench-Long official prompt set on 30-second video generation following LongLive.

#### 4.1 COMPARISONS WITH BASELINES

**Quantitative Comparisons.** As shown in Tab. 1, Mask Forcing improves all baseline methods on both visual quality (HPSv3 & Vision.) and semantic score (Instruct.) on the 100-prompt set under both chunk-wise and frame-wise settings. Especially for visual quality, the consistent improvements in visual quality align with our motivation of perturbing student rollout trajectories to alleviate artifacts associated with mode collapse. For motion evaluation, incorporating Mask Forcing produces higher motion quality (MQ) and dynamic degree (Dynamic.). Though the dynamic degree score for LongLive is lower, this can be attributed to generic VBench rewarding drift-induced optical flow (Minar et al., 2026). For the VBench benchmark, including Mask Forcing surpasses all base-

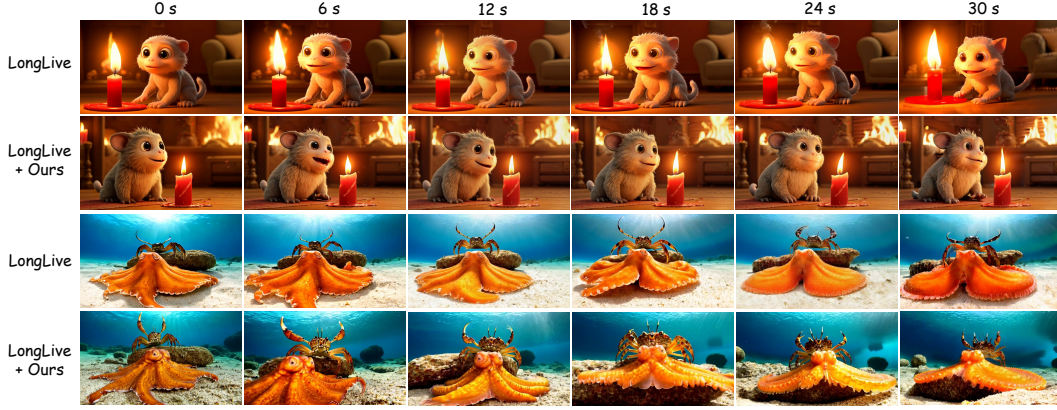


Figure 4: Qualitative results of single-prompt long-video setting on LongLive. Incorporating our method (+Ours) enhances the video generation quality with more visual details.

Table 2: Single-prompt 30s long video generation on MovieGen and VBench-Long benchmarks.

| Method   | HPSv3 $\uparrow$            | Vision. $\uparrow$           | Instruct. $\uparrow$          | MQ $\uparrow$                | Dynamic. $\uparrow$     | Total $\uparrow$             | Quality $\uparrow$           | Semantic $\uparrow$          |
|----------|-----------------------------|------------------------------|-------------------------------|------------------------------|-------------------------|------------------------------|------------------------------|------------------------------|
| LongLive | 8.44                        | 14.31                        | 62.04                         | 18.84                        | <b>70</b>               | 83.91                        | 84.71                        | 80.70                        |
| + Ours   | <b>9.11</b> <sub>+.67</sub> | <b>14.93</b> <sub>+.62</sub> | <b>66.02</b> <sub>+.398</sub> | <b>19.37</b> <sub>+.53</sub> | <b>64</b> <sub>-6</sub> | <b>84.51</b> <sub>+.60</sub> | <b>85.28</b> <sub>+.57</sub> | <b>81.44</b> <sub>+.74</sub> |

line methods across the VBench metrics in both settings. Additionally, we evaluate Mask Forcing on single-prompt long video generation with LongLive, integrating it into both the initialization and streaming long-video tuning stages. As reported in Tab. 2, our method can also effectively improve visual quality in the long video generation setting.

**Qualitative Comparisons.** Fig. 3 presents the qualitative comparisons of our method and the baselines on the 100-prompt set and MovieGen. The baseline methods commonly exhibit limited visual quality, with over-saturation and over-smoothing artifacts. Specifically, Self Forcing produces the woman’s face with unnaturally high-contrast coloration and the creature with a visually flat appearance, while LongLive generates scenes lacking fine-grained details. The rock-man and kitchen examples produced by Causal Forcing also exhibit severe over-saturation. Incorporating Mask Forcing effectively alleviates these issues, greatly enhancing the visual quality and realism of the generated videos. We additionally present the qualitative results for long video generation in Fig. 4. Incorporating Mask Forcing produces higher visual quality with more high-frequency details, such as the fluffy fur and candle in case 1 and the fine-grained stone and sand textures in case 2.

#### 4.2 ABLATION STUDIES

We conduct comprehensive ablation studies on the 100-prompt set with Self Forcing as the baseline to validate the design choices in Mask Forcing. We jointly report HPSv3 and Dynamic Degree to measure visual quality and motion dynamics during analysis since models trained with DMD may suffer from reduced motion dynamics as training progresses. We further compare convergence speed across different baselines with and without Mask Forcing.

**Mask Ratio  $\alpha$ .** We compare different mask ratio values for the dual-noise masking rollout in Tab. 3. With a small ratio of 0.1, the masking trajectories only induce similar rollouts and provide limited exploration beyond the modes already covered by the student. Therefore, it achieves a high HPSv3 score of 10.00 but exhibits weak motion dynamics, with a Dynamic Degree of 47. With large ratios of 0.4-0.5, a substantial proportion of the input consists of cleaner states derived from the current student prediction. The resulting strong denoising cues yield the highest HPSv3 scores of 10.15 and 10.17. However, the Dynamic Degree decreases to 57 and 44, suggesting that excessive reliance on cleaner predictions may restrict the model’s ability to revise or diversify motion patterns during subsequent denoising. A moderate mask ratio of 0.2 introduces sufficient perturbation to diversify the student rollout trajectories while avoiding excessive reliance on cleaner states derived from the



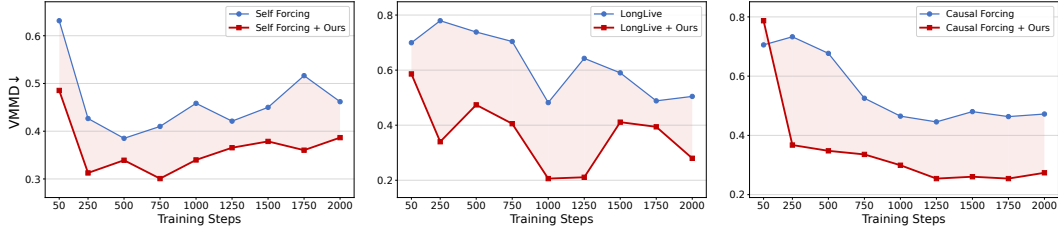


Figure 5: V-JEPA2 Maximum Mean Discrepancy (VMMD) over training steps on the evaluation set. Incorporating Mask Forcing leads to faster convergence across various baselines.

Table 3: Mask ratio  $\alpha$  ablation.

| $\alpha$ | HPSv3 $\uparrow$ | Dynamic. $\uparrow$ |
|----------|------------------|---------------------|
| 0.1      | 10.00            | 47                  |
| 0.2      | 9.84             | 82                  |
| 0.3      | 9.82             | 65                  |
| 0.4      | 10.15            | 57                  |
| 0.5      | 10.17            | 44                  |

Table 4: Timestep window  $\Delta$  ablation.

| $\Delta$ | HPSv3 $\uparrow$ | Dynamic. $\uparrow$ |
|----------|------------------|---------------------|
| 50       | 9.16             | 92                  |
| 150      | 9.82             | 70                  |
| 250      | 9.84             | 82                  |
| 450      | 9.98             | 80                  |
| 600      | 9.90             | 67                  |

Table 5: Mask scheme ablation.

| Mask scheme            | HPSv3 $\uparrow$ | Dynamic. $\uparrow$ |
|------------------------|------------------|---------------------|
| shared, per-rollout    | 10.12            | 56                  |
| shared, per-chunk      | 9.84             | 81                  |
| shared, per-step       | 9.64             | 72                  |
| per-frame, per-rollout | 10.03            | 53                  |
| per-frame, per-chunk   | 9.84             | 82                  |
| per-frame, per-step    | 9.93             | 74                  |

current student prediction. It achieves a competitive HPSv3 score of 9.84 and the highest Dynamic Degree of 82, providing a balance between visual quality and motion dynamics.

**Timestep Window  $\Delta$ .** We ablate the timestep window  $\Delta$  used to sample the timestep corresponding to a lower noise level in Tab. 4. This window controls the maximum difference between original and cleaner noise levels. With a small timestep window of  $\Delta = 50$ , the cleaner noise level remains close to the original, providing only mild rollout perturbations and limited denoising cues. Therefore, this setting achieves a high Dynamic Degree of 92 but a low HPSv3 score of 9.16. Increasing  $\Delta$  strengthens the rollout perturbation and provides cleaner tokens with more informative denoising cues. Consequently, increasing  $\Delta$  to 150, 250, and 450 improves the HPSv3 score to 9.82, 9.84, 9.98, respectively. However, an excessively large window of  $\Delta = 600$  may cause cleaner predictions to dominate subsequent denoising, limiting motion diversity. Therefore, we adopt  $\Delta = 250$ , which achieves a competitive HPSv3 score of 9.84 while maintaining a high Dynamic Degree of 82.

**Mask Scheme.** We ablate the influence of different mask schemes for chunk-wise generation in Tab. 5. We evaluate the schemes varied along spatial and temporal axes: across the frames within each chunk and across chunks during the rollout. Spatially, “shared” applies the same mask to all frames within a chunk and “per-frame” independently samples a mask for each frame. Temporally, “per-rollout” keeps the mask fixed throughout the entire rollout, “per-chunk” resamples the mask for each chunk while keeping it fixed across the denoising steps within the chunk, and “per-step” resamples the mask at every denoising step. These spatial and temporal mask choices determine how the mask-induced perturbations vary throughout the rollout and consequently affect the resulting student rollout distributions. An effective scheme should sufficiently diversify the student distribution to improve its coverage of the teacher modes without introducing excessive variation that deviates from the teacher distribution. As shown in Tab. 5, applying “per-frame” in the spatial axis and “per-chunk” in the temporal axis achieves both high visual quality and motion dynamics, with a HPSv3 score of 9.84 and Dynamic Degree of 82.

**Convergence Speed.** We compare the training convergence of different baselines with and without Mask Forcing in Figs. 1, 5 and 9. We use HPSv3 to evaluate the visual quality trend throughout the training. To assess the distributional alignment between student- and teacher-generated videos, we additionally report Maximum Mean Discrepancy (MMD) (Jayasumana et al., 2024) in the CLIP (Radford et al., 2021) and V-JEPA2 (Assran et al., 2025) feature spaces, denoted as CMMD and VMMD, respectively. Specifically, we first generate a fixed reference set with the real-score teacher on the 100-prompt set. We then generate videos for each step using the causal student on the same prompts and compute CMMD and VMMD against the teacher reference set. As shown

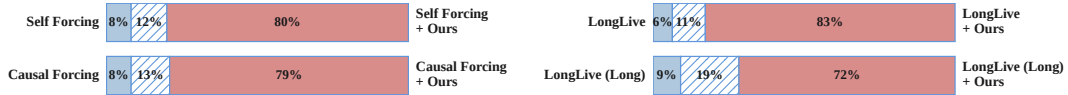


Figure 6: Pairwise human preferences between baseline models with and without Mask Forcing. Blue and red denote preferences for models without and with Mask Forcing, respectively, while hatched regions indicate ties.

in Figs. 1, 5 and 9, Mask Forcing accelerates convergence for all baselines in terms of both visual quality and distributional alignment.

### 4.3 HUMAN EVALUATION

We conduct pairwise human evaluations between anonymized outputs from Mask Forcing (+Ours) against different baselines for short and long video generation. We recruit 24 users to conduct this human evaluation study, and each user is required to select the video with higher visual quality and greater realism from each pair of videos. As shown in Fig. 6, incorporating Mask Forcing receives 80%, 79%, and 83% of the preference votes over the baseline methods Self Forcing, Causal Forcing, and LongLive, respectively. Moreover, for long video generation, Mask Forcing receives 72% of the preference votes over the LongLive baseline, further demonstrating the effectiveness of our method.

## 5 CONCLUSION

In this paper, we introduce Mask Forcing, a Dual-Noise Masking Rollout strategy for AR video diffusion distillation that alleviates the mode collapse induced by the mode-seeking reverse KL objective in DMD. Mask Forcing injects lower-noise signals into the noisy rollout inputs via random masks varying along spatial and temporal axes. These perturbations promote broader exploration during self-rollout, diversifying student trajectories and encouraging the student distribution to cover more regions of the teacher distribution and thereby allowing DMD to provide learning signals beyond the modes already covered by the student. Meanwhile, cleaner tokens act as denoising guidance for noisier tokens, improving intermediate rollout predictions and mitigating error accumulation during the self-rollout process. Extensive experiments demonstrate the effectiveness of our method across multiple baselines, with improved visual quality and convergence speed. We further investigate the effects of different masking mechanisms on the visual quality and motion dynamics balance on the generated videos. Overall, Mask Forcing provides a simple and effective strategy to improve AR video diffusion distillation without incorporating real video data or post-training.

## REFERENCES

- Mido Assran, Adrien Bardes, David Fan, Quentin Garrido, Russell Howes, Matthew Muckley, Ammar Rizvi, Claire Roberts, Koustuv Sinha, Artem Zhoulus, et al. V-jepa 2: Self-supervised video models enable understanding, prediction and planning. *arXiv preprint arXiv:2506.09985*, 2025.
- Hangbo Bao, Li Dong, Songhao Piao, and Furu Wei. Beit: Bert pre-training of image transformers. *arXiv preprint arXiv:2106.08254*, 2021.
- Shengqu Cai, Weili Nie, Chao Liu, Julius Berner, Lvmin Zhang, Nanye Ma, Hansheng Chen, Maneesh Agrawala, Leonidas Guibas, Gordon Wetzstein, et al. Mode seeking meets mean seeking for fast long video generation. *arXiv preprint arXiv:2602.24289*, 2026.
- Huiwen Chang, Han Zhang, Lu Jiang, Ce Liu, and William T Freeman. Maskgit: Masked generative image transformer. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 11315–11325, 2022.
- Hila Chefer, Patrick Esser, Dominik Lorenz, Dustin Podell, Vikash Raja, Vinh Tong, Antonio Torralba, and Robin Rombach. Self-supervised flow matching for scalable multi-modal synthesis. *arXiv preprint arXiv:2603.06507*, 2026.
- Boyuan Chen, Diego Martí Monsó, Yilun Du, Max Simchowitz, Russ Tedrake, and Vincent Sitzmann. Diffusion forcing: Next-token prediction meets full-sequence diffusion. *Advances in Neural Information Processing Systems*, 37:24081–24125, 2024.
- Howard Chen, Noam Razin, Karthik Narasimhan, and Danqi Chen. Retaining by doing: The role of on-policy data in mitigating forgetting. *arXiv preprint arXiv:2510.18874*, 2025.
- Siyi Chen, Shaowei Liu, Yixuan Jia, Zian Wang, Huan Ling, Qing Qu, and Jun Gao. Data-forcing distillation: Restoring diversity and fidelity in few-step video generation. *arXiv preprint arXiv:2606.18478*, 2026.
- Kaifeng Gao, Jiabin Shi, Hanwang Zhang, Chunping Wang, Jun Xiao, and Long Chen. Ca2-vdm: Efficient autoregressive video diffusion model with causal generation and cache sharing. In *ICML*. PMLR, 2025.
- Zelin Gao, Qiuyu Wang, Jiapeng Zhu, Jingye Chen, Zichen Liu, Qingyan Bai, Jiahao Wang, Yufeng Yuan, Hanlin Wang, Yichong Lu, Ka Leong Cheng, Haojie Zhang, Jian Gao, Tianrui Feng, Yuzheng Liu, Yao Yao, Yinghao Xu, Xing Zhu, Yujun Shen, and Hao Ouyang. Infinite worlds with versatile interactions. *arXiv preprint arXiv:2607.07534*, 2026.
- Yoav HaCohen, Benny Brazowski, Nisan Chiprut, Yaki Bitterman, Andrew Kvochko, Avishai Berkowitz, Daniel Shalem, Daphna Lifschitz, Dudu Moshe, Eitan Porat, et al. Ltx-2: Efficient joint audio-visual foundation model. *arXiv preprint arXiv:2601.03233*, 2026.
- Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 16000–16009, 2022.
- Anthony Hu, Václav Volhejn, Adrien Ramanana Rahary, Chris Mulder, Aditya Makkar, Amélie Royer, Manu Orsini, Alyx Liao, Adam Jelley, Eloi Alonso, et al. Multiplayer interactive world models with representation autoencoders. *arXiv preprint arXiv:2607.05352*, 2026.
- Jinyi Hu, Shengding Hu, Yuxuan Song, Yufei Huang, Mingxuan Wang, Hao Zhou, Zhiyuan Liu, Wei-Ying Ma, and Maosong Sun. Acddit: Interpolating autoregressive conditional modeling and diffusion transformer. *arXiv preprint arXiv:2412.07720*, 2024.
- Junchao Huang, Guian Fang, Shengju Qian, Xianghao Kong, Zhuoran Zhao, Wei Huang, Yihua Du, Zixin Zhang, Justin Cui, Yuchao Gu, et al. Solarwm: Open data and scalable training for long-horizon video world models. *arXiv preprint arXiv:2609.02886*, 2026a.
- Xun Huang, Zhengqi Li, Guande He, Mingyuan Zhou, and Eli Shechtman. Self forcing: Bridging the train-test gap in autoregressive video diffusion. *Advances in Neural Information Processing Systems*, 38:167283–167308, 2026b.

- Team HunyuanWorld. Hy-world 1.5: A systematic framework for interactive world modeling with real-time latency and geometric consistency. *arXiv preprint*, 2025.
- Sadeep Jayasumana, Srikumar Ramalingam, Andreas Veit, Daniel Glasner, Ayan Chakrabarti, and Sanjiv Kumar. Rethinking fid: Towards a better evaluation metric for image generation. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9307–9315. IEEE, 2024.
- Xuan Ju, Yiming Gao, Zhaoyang Zhang, Ziyang Yuan, Xintao Wang, Ailing Zeng, Yu Xiong, Qiang Xu, and Ying Shan. Miradata: A large-scale video dataset with long durations and structured captions. *Advances in Neural Information Processing Systems*, 37:48955–48970, 2024.
- Dongjun Kim, Chieh-Hsin Lai, WeiHsiang 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 *International Conference on Learning Representations*, volume 2024, pp. 44493–44525, 2024.
- Weijie Kong, Qi Tian, Zijian Zhang, Rox Min, Zuozhuo Dai, Jin Zhou, Jiangfeng Xiong, Xin Li, Bo Wu, Jianwei Zhang, et al. Hunyuanvideo: A systematic framework for large video generative models. *arXiv preprint arXiv:2412.03603*, 2024.
- Jiaxing Li, Kai Zou, Cindy Zhou, Kaichen Huang, Junyao Gao, Zile Wang, Yang Liu, Bin Liu, Bo An, and Yangguang Li. Distillalign: Coordinating mode covering and mode seeking in autoregressive video distillation. *arXiv preprint arXiv:2607.26811*, 2026a.
- Zhen Li, Chuanhao Li, Xiaofeng Mao, Shaoheng Lin, Ming Li, Shitian Zhao, Zhaopan Xu, Xinyue Li, Yukang Feng, Jianwen Sun, et al. Sekai: A video dataset towards world exploration. *Advances in Neural Information Processing Systems*, 38, 2026b.
- Lu Ling, Yichen Sheng, Zhi Tu, Wentian Zhao, Cheng Xin, Kun Wan, Lantao Yu, Qianyu Guo, Zixun Yu, Yawen Lu, et al. DI3dv-10k: A large-scale scene dataset for deep learning-based 3d vision. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 22160–22169. IEEE, 2024.
- Hongyu Liu, Chun Wang, Feng Gao, Xuanhua He, Yue Ma, Ziyu Wan, Yong Zhang, Xiaoming Wei, and Qifeng Chen. Opsd-v: On-policy self-distillation for post-training few-step autoregressive video generators. *arXiv preprint arXiv:2607.08766*, 2026.
- Cheng Lu and Yang Song. Simplifying, stabilizing and scaling continuous-time consistency models. In *International Conference on Learning Representations*, volume 2025, pp. 50611–50649, 2025.
- Yuhang Ma, Xiaoshi Wu, Keqiang Sun, and Hongsheng Li. Hpsv3: Towards wide-spectrum human preference score. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 15086–15095, 2025.
- Matiur Rahman Minar, Seunghun Oh, Ganghyeon Jeong, and Unsang Park. Steady-forcing: Balancing spatial persistence and motion continuity in long-horizon nature video diffusion. *arXiv preprint arXiv:2606.14732*, 2026.
- 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.
- Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pp. 8748–8763. PmLR, 2021.
- Oriane Siméoni, Huy V Vo, Maximilian Seitzer, Federico Baldassarre, Maxime Oquab, Cijo Jose, Vasil Khalidov, Marc Szafraniec, Seungeun Yi, Michaël Ramamonjisoa, et al. Dinov3. *arXiv preprint arXiv:2508.10104*, 2025.
- Yang Song, Prafulla Dhariwal, Mark Chen, and Ilya Sutskever. Consistency models. *arXiv preprint arXiv:2303.01469*, 2023.



- DreamX Team, Yancheng Bai, Rui Chen, Xiangxiang Chu, Rujing Dang, Hao Dou, Bingjie Gao, Qiwen Gu, Siyu Hong, Jiachen Lei, et al. Dreamx-world 1.0: A general-purpose interactive world model. *arXiv preprint arXiv:2606.16993*, 2026a.
- Robbyant Team, Zelin Gao, Qiuyu Wang, Yanhong Zeng, Jiapeng Zhu, Ka Leong Cheng, Yixuan Li, Hanlin Wang, Yinghao Xu, Shuailei Ma, et al. Advancing open-source world models. *arXiv preprint arXiv:2601.20540*, 2026b.
- Zachary Teed and Jia Deng. Raft: Recurrent all-pairs field transforms for optical flow. In *European conference on computer vision*, pp. 402–419. Springer, 2020.
- Team Wan, Ang Wang, Baole Ai, Bin Wen, Chaojie Mao, Chen-Wei Xie, Di Chen, Feiwei Yu, Haiming Zhao, Jianxiao Yang, Jianyuan Zeng, Jiayu Wang, Jingfeng Zhang, Jingren Zhou, Jinkai Wang, Jixuan Chen, Kai Zhu, Kang Zhao, Keyu Yan, Lianghua Huang, Mengyang Feng, Ningyi Zhang, Pandeng Li, Pingyu Wu, Ruihang Chu, Ruili Feng, Shiwei Zhang, Siyang Sun, Tao Fang, Tianxing Wang, Tianyi Gui, Tingyu Weng, Tong Shen, Wei Lin, Wei Wang, Wei Wang, Wenmeng Zhou, Wenten Wang, Wenting Shen, Wenyuan Yu, Xianzhong Shi, Xiaoming Huang, Xin Xu, Yan Kou, Yangyu Lv, Yifei Li, Yijing Liu, Yiming Wang, Yingya Zhang, Yitong Huang, Yong Li, You Wu, Yu Liu, Yulin Pan, Yun Zheng, Yuntao Hong, Yupeng Shi, Yutong Feng, Zeyinzi Jiang, Zhen Han, Zhi-Fan Wu, and Ziyu Liu. Wan: Open and advanced large-scale video generative models. *arXiv preprint arXiv:2503.20314*, 2025.
- Jiahao Wang, Yufeng Yuan, Rujie Zheng, Youtian Lin, Jian Gao, Lin-Zhuo Chen, Yajie Bao, Chang Zeng, Yanxi Zhou, Xiao-Xiao Long, et al. Spatialvid: A large-scale video dataset with spatial annotations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 42592–42603, 2026a.
- Wenhao Wang and Yi Yang. Vidprom: A million-scale real prompt-gallery dataset for text-to-video diffusion models. *Advances in Neural Information Processing Systems*, 37:65618–65642, 2024.
- Zile Wang, Zexiang Liu, Jiaxing Li, Kaichen Huang, Baixin Xu, Fei Kang, Mengyin An, Peiyu Wang, Biao Jiang, Yichen Wei, et al. Matrix-game 3.0: Real-time and streaming interactive world model with long-horizon memory. *arXiv preprint arXiv:2604.08995*, 2026b.
- Tianhe Wu, Ruibin Li, Lei Zhang, and Kede Ma. Diversity-preserved distribution matching distillation for fast visual synthesis. In *Forty-third International Conference on Machine Learning*, 2026. URL <https://openreview.net/forum?id=R3JzLU2qZj>.
- Jiazheng Xu, Yu Huang, Jiale Cheng, Yuanming Yang, Jiajun Xu, Yuan Wang, Wenbo Duan, Shen Yang, Qunlin Jin, Shurun Li, et al. Visionreward: Fine-grained multi-dimensional human preference learning for image and video generation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 11269–11277, 2026.
- Shuai Yang, Wei Huang, Ruihang Chu, Yicheng Xiao, Yuyang Zhao, Xianbang Wang, Muyang Li, Enze Xie, Yingcong Chen, Yao Lu, et al. Longlive: Real-time interactive long video generation. *arXiv preprint arXiv:2509.22622*, 2025.
- 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*, 2024a.
- Tianwei Yin, Michaël Gharbi, Richard Zhang, Eli Shechtman, Fredo Durand, William T Freeman, and Taesung Park. One-step diffusion with distribution matching distillation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 6613–6623, 2024b.
- Songchun Zhang, Zeyue Xue, Siming Fu, Jie Huang, Xianghao Kong, Y Ma, Haoyang Huang, Nan Duan, and Anyi Rao. Astrolabe: Steering forward-process reinforcement learning for distilled autoregressive video models. *arXiv preprint arXiv:2603.17051*, 2026.
- Min Zhao, Hongzhou Zhu, Bokai Yan, Zihan Zhou, Yimin Chen, Wenqiang Sun, Kaiwen Zheng, Guande He, Xiao Yang, Chongxuan Li, et al. minwm: A full-stack open-source framework for real-time interactive video world models. *arXiv preprint arXiv:2605.30263*, 2026.

Guangcong Zheng, Teng Li, Xianpan Zhou, and Xi Li. Realcam-vid: High-resolution video dataset with dynamic scenes and metric-scale camera movements. *arXiv preprint arXiv:2504.08212*, 2025. URL <https://arxiv.org/abs/2504.08212>.

Kaiwen Zheng, Guande He, Min Zhao, Jintao Zhang, Huayu Chen, Jianfei Chen, Chen-Hsuan Lin, Ming-Yu Liu, Jun Zhu, and Qianli Ma. Causal-rcm: A unified teacher-forcing and self-forcing open recipe for autoregressive diffusion distillation in streaming video generation and interactive world models. *arXiv preprint arXiv:2606.25473*, 2026a.

Kaiwen Zheng, Yuji Wang, Qianli Ma, Huayu Chen, Jintao Zhang, Yogesh Balaji, Jianfei Chen, Ming-Yu Liu, Jun Zhu, and Qinsheng Zhang. Large scale diffusion distillation via score-regularized continuous-time consistency. In *The Fourteenth International Conference on Learning Representations*, 2026b. URL <https://openreview.net/forum?id=2uNlM353RI>.

Yang Zhou, Yifan Wang, Jianjun Zhou, Wenzheng Chang, Haoyu Guo, Zizun Li, Kaijing Ma, Xinyue Li, Yating Wang, Haoyi Zhu, et al. Omniworld: A multi-domain and multi-modal dataset for 4d world modeling. *arXiv preprint arXiv:2509.12201*, 2025.

Hongzhou Zhu, Min Zhao, Guande He, Hang Su, Chongxuan Li, and Jun Zhu. Causal forcing: Autoregressive diffusion distillation done right for high-quality real-time interactive video generation. *arXiv preprint arXiv:2602.02214*, 2026.

Junhao Zhuang, Shiyi Zhang, Yuxuan Bian, Yaowei Li, Yawen Luo, Yijun Liu, Weiyang Jin, Songchun Zhang, Xianglong He, Xuying Zhang, Haoran Li, Haoyang Huang, Zeyue Xue, and Nan Duan. Self gradient forcing: Native long video extrapolation, 2026. URL <https://arxiv.org/abs/2607.20368>.

## 6 APPENDIX

### 6.1 THEORETICAL JUSTIFICATION

In this section, we characterize the distribution induced by Mask Forcing and derive an exact decomposition of its reverse-KL objective. Fix a text condition  $c$  and a DMD score-noising timestep  $\tau > 0$ . Let

$$p_\tau(x) := p_{\text{real},\tau}(x | c) \quad (14)$$

denote the teacher noisy marginal. Let

$$V := \{(M_r, t_{k,r})\}_{r=1}^R, \quad V \sim \pi, \quad (15)$$

collect the masks and cleaner timesteps sampled during one Mask Forcing rollout, where  $\pi$  is the distribution induced by the corresponding samplers.

Given a masking trajectory  $V = v$ , define the conditional student distribution as

$$q_{\theta,\tau}^v(x) := p_\theta(X_\tau = x | V = v, c). \quad (16)$$

Marginalizing over  $V$  gives

$$\bar{q}_{\theta,\tau}(x) := \mathbb{E}_{V \sim \pi} [q_{\theta,\tau}^V(x)]. \quad (17)$$

Under exact score estimation, the real and fake scores at noise level  $\tau$  are

$$s_{\text{real}}(x, \tau, c) = \nabla_x \log p_\tau(x), \quad s_{\text{fake}}(x, \tau, c) = \nabla_x \log \bar{q}_{\theta,\tau}(x). \quad (18)$$

Since the fake-score model is not conditioned on  $V$ , it estimates the score of the marginal distribution  $\bar{q}_{\theta,\tau}$  rather than a component score  $\nabla_x \log q_{\theta,\tau}^v$ .

The DMD score-difference update corresponds to the gradient

$$\nabla_\theta D_{\text{KL}}(\bar{q}_{\theta,\tau} \| p_\tau) = \mathbb{E} \left[ (s_{\text{fake}}(X_\tau, \tau, c) - s_{\text{real}}(X_\tau, \tau, c))^\top \nabla_\theta X_\tau \right]. \quad (19)$$

We therefore analyze the reverse-KL objective

$$D_{\text{KL}}(\bar{q}_{\theta,\tau} \| p_\tau). \quad (20)$$

The exact-score assumption is used only for this identification. In practice, both score networks approximate the corresponding marginal scores.

**Proposition (KL decomposition for the masked-rollout distribution).** Suppose that the KL divergences below are finite. Then

$$D_{\text{KL}}(\bar{q}_{\theta,\tau} \| p_\tau) = \mathbb{E}_V [D_{\text{KL}}(q_{\theta,\tau}^V \| p_\tau)] - I_\theta(V; X_\tau | c). \quad (21)$$

**Proof.** By the definition of KL divergence, the average conditional KL can be expanded by multiplying and dividing the density ratio by  $\bar{q}_{\theta,\tau}$ :

$$\begin{aligned} & \mathbb{E}_V D_{\text{KL}}(q_{\theta,\tau}^V \| p_\tau) \\ &= \int \pi(v) \int q_{\theta,\tau}^v(x) \log \frac{q_{\theta,\tau}^v(x)}{p_\tau(x)} dx dv \\ &= \mathbb{E}_{V, X_\tau | c} \left[ \log \frac{q_{\theta,\tau}^V(X_\tau)}{p_\tau(X_\tau)} \right] \\ &= \mathbb{E}_{V, X_\tau | c} \left[ \log \frac{q_{\theta,\tau}^V(X_\tau)}{\bar{q}_{\theta,\tau}(X_\tau)} + \log \frac{\bar{q}_{\theta,\tau}(X_\tau)}{p_\tau(X_\tau)} \right] \\ &= \mathbb{E}_{V, X_\tau | c} \left[ \log \frac{q_{\theta,\tau}^V(X_\tau)}{\bar{q}_{\theta,\tau}(X_\tau)} \right] + \mathbb{E}_{V, X_\tau | c} \left[ \log \frac{\bar{q}_{\theta,\tau}(X_\tau)}{p_\tau(X_\tau)} \right] \\ &= \underbrace{\mathbb{E}_V D_{\text{KL}}(q_{\theta,\tau}^V \| \bar{q}_{\theta,\tau})}_{I_\theta(V; X_\tau | c)} + D_{\text{KL}}(\bar{q}_{\theta,\tau} \| p_\tau). \end{aligned} \quad (22)$$

The first term is the conditional mutual information. By definition,

$$\begin{aligned} I_\theta(V; X_\tau | c) &= \mathbb{E}_{V, X_\tau | c} \left[ \log \frac{p_\theta(V, X_\tau | c)}{p(V | c)p_\theta(X_\tau | c)} \right] \\ &= \int \pi(v) q_{\theta,\tau}^v(x) \log \frac{\pi(v) q_{\theta,\tau}^v(x)}{\pi(v) \bar{q}_{\theta,\tau}(x)} dx dv \\ &= \int \pi(v) \left[ \int q_{\theta,\tau}^v(x) \log \frac{q_{\theta,\tau}^v(x)}{\bar{q}_{\theta,\tau}(x)} dx \right] dv \\ &= \int \pi(v) D_{\text{KL}}(q_{\theta,\tau}^v \| \bar{q}_{\theta,\tau}) dv \\ &= \mathbb{E}_V D_{\text{KL}}(q_{\theta,\tau}^V \| \bar{q}_{\theta,\tau}). \end{aligned} \quad (23)$$

For the second term, we first expand the joint expectation and then marginalize over  $V$ :

$$\begin{aligned} & \mathbb{E}_{V, X_\tau | c} \left[ \log \frac{\bar{q}_{\theta,\tau}(X_\tau)}{p_\tau(X_\tau)} \right] \\ &= \int \pi(v) \left[ \int q_{\theta,\tau}^v(x) \log \frac{\bar{q}_{\theta,\tau}(x)}{p_\tau(x)} dx \right] dv \\ &= \int \left[ \int \pi(v) q_{\theta,\tau}^v(x) dv \right] \log \frac{\bar{q}_{\theta,\tau}(x)}{p_\tau(x)} dx \\ &= \int \bar{q}_{\theta,\tau}(x) \log \frac{\bar{q}_{\theta,\tau}(x)}{p_\tau(x)} dx \\ &= D_{\text{KL}}(\bar{q}_{\theta,\tau} \| p_\tau). \end{aligned} \quad (24)$$

Combining the two terms and rearranging proves equation 21.

The first term in equation 21 is the average reverse KL of the conditional rollout distributions. The mutual information measures their dependence on the masking trajectory. It also follows from equation 21 that

$$D_{\text{KL}}(\bar{q}_{\theta,\tau} \| p_\tau) \leq \mathbb{E}_V D_{\text{KL}}(q_{\theta,\tau}^V \| p_\tau). \quad (25)$$

Since mutual information is nonnegative, equality holds when  $V$  and  $X_\tau$  are conditionally independent given  $c$ , meaning that the masking trajectory does not affect the conditional output distribution. If different masking trajectories induce distinguishable conditional output distributions, then  $I_\theta(V; X_\tau | c) > 0$ , and the inequality is strict.

Table 6: Quantitative comparison of Mask Forcing (Ours) with Causal-rCM and DistillAlign on the 100-prompt set benchmark.

| Method                           | HPSv3 $\uparrow$ | Vision. $\uparrow$ | Instruct. $\uparrow$ | MQ $\uparrow$ | Dynamic. $\uparrow$ |
|----------------------------------|------------------|--------------------|----------------------|---------------|---------------------|
| DistillAlign (Li et al., 2026a)  | 9.29             | 9.46               | 37.69                | 14.36         | 76                  |
| Causal-rCM (Zheng et al., 2026a) | 9.61             | 9.43               | 38.50                | 14.34         | 64                  |
| Ours                             | <b>10.17</b>     | <b>11.58</b>       | <b>46.30</b>         | <b>21.54</b>  | <b>82</b>           |

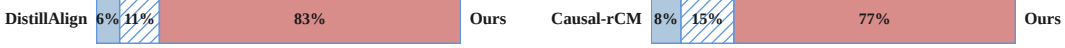


Figure 7: Pairwise human preference between joint distillation methods (DistillAlign and Causal-rCM) and Mask Forcing.

**Implication for mode coverage.** Under limited model capacity, reverse-KL optimization may concentrate a conditional rollout distribution on a high-density region of the teacher distribution. If different masking trajectories induce distinct conditional distributions, then  $I_\theta(V; X_\tau | c) > 0$ , and equation 21 shows that their marginal mixture has strictly lower reverse KL than their average conditional reverse KL. This mixture can cover multiple teacher-supported regions without requiring each conditional distribution to place probability mass between them. The result therefore gives a distributional mechanism by which randomized masking can alleviate mode-seeking behavior.

## 6.2 COMPARISON WITH JOINT DISTILLATION

Recent methods seek to improve distillation performance by combining complementary objectives that balance mode-seeking and mode-covering behaviors, such as rCM and DistillAlign. We compare with Causal-rCM (Zheng et al., 2026a) and DistillAlign in terms of visual quality and motion dynamics on the 100-prompt set benchmark. While rCM jointly optimizes continuous-time consistency distillation and score-based distillation in bidirectional models, Causal-rCM decouples these objectives into two stages. It first performs consistency distillation under teacher forcing to obtain causal initialization with broad mode coverage, and then applies DMD under self-forcing. Since DistillAlign is built upon Causal Forcing and Causal-rCM pairs its two distillation objectives with two causal training paradigms, we compare Mask Forcing on the Causal Forcing baseline with them. For DistillAlign, we use its final joint-distilled generator with the Wan2.1-T2V-14B teacher. For Causal-rCM, we follow its 4-step inference setting.

As shown in Tab. 6, our method outperforms both methods in visual quality, motion dynamics, as well as semantic alignment. Qualitative comparison in Fig. 8 is consistent with the quantitative results, showing that our method generates videos with richer details and greater realism. We further conduct pairwise human preference against joint distillation methods DistillAlign and Causal-rCM. As shown in Fig. 7, Mask Forcing receives more preference votes, with 83% and 77% over DistillAlign and Causal-rCM, respectively. These results demonstrate that Mask Forcing yields videos with higher visual quality and realism from the users’ perspective.

## 6.3 ADDITIONAL EVALUATION ON LONGLIVE

We compare the convergence of LongLive with and without Mask Forcing in Fig. 9. Consistent with the results on Self Forcing and Causal Forcing, Mask Forcing achieves higher HPSv3 and lower CMMD in fewer training steps, demonstrating faster convergence.

Moreover, we evaluate LongLive and Mask Forcing on MovieGen over consecutive 6-second intervals from 0 to 30 seconds. For each interval, we average the CLIP score over all frames and the HPSv3 score over 12 uniformly sampled frames. As reported in Tab. 8, both methods exhibit a certain decline in CLIP and HPSv3 scores as the video generation progresses, likely due to error accumulation during autoregressive self-rollout. However, incorporating Mask Forcing achieves higher CLIP and HPSv3 scores than LongLive across all intervals, demonstrating consistent improvements in semantic alignment and visual quality throughout long video generation.





Figure 8: Qualitative comparison of our method with Causal-rCM and DistillAlign. Our method produces videos with better visual quality and motion dynamics.

#### 6.4 DIVERSITY EVALUATION

We further evaluate the video diversity across different baselines under the chunk-wise setting using CLIP-ViT-Large (Radford et al., 2021) and DINOv3-ViT-Large (Siméoni et al., 2025) following DP-DMD (Wu et al., 2026). We use the 100-prompt set as the evaluation set and generate 8 videos for each prompt with random seeds 0-7, generating 800 videos for each baseline for evaluation in total.

As shown in Tab. 7, Mask Forcing improves CLIP diversity across all three baselines, indicating consistently greater semantic variation. It also substantially improves DINOv3 diversity for Causal Forc-

Table 7: Video diversity across baselines.

| Method         | CLIP Div.↑    | DINO Div.↑    |
|----------------|---------------|---------------|
| Self Forcing   | 0.0773        | <b>0.1704</b> |
| + Ours         | <b>0.0814</b> | 0.1645        |
| LongLive       | 0.0784        | 0.1537        |
| + Ours         | <b>0.0819</b> | <b>0.1750</b> |
| Causal Forcing | 0.0675        | 0.1371        |
| + Ours         | <b>0.0720</b> | <b>0.1577</b> |



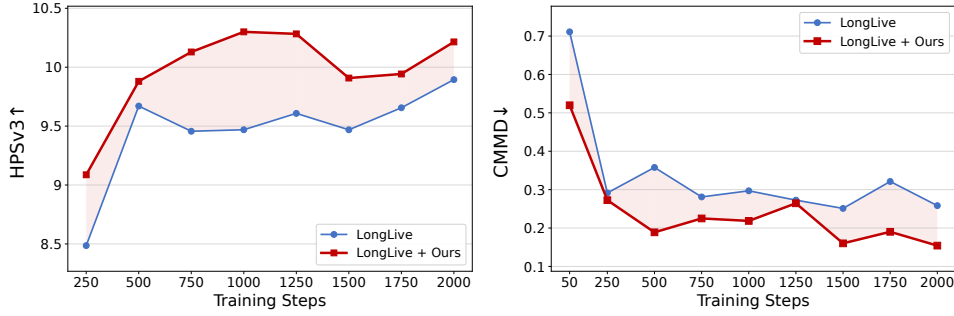


Figure 9: HPSv3 and CMMD over training steps on the evaluation set for LongLive and LongLive (+Ours).

Table 8: Quantitative evaluation on long video generation on MovieGen. We evaluate the CLIP Score and HPSv3 across 6-second intervals (0–30s).

| Method   | CLIP↑        |              |              |              |              | HPSv3↑      |             |             |             |             |
|----------|--------------|--------------|--------------|--------------|--------------|-------------|-------------|-------------|-------------|-------------|
|          | 0-6          | 6-12         | 12-18        | 18-24        | 24-30        | 0-6         | 6-12        | 12-18       | 18-24       | 24-30       |
| LongLive | 33.70        | 33.41        | 33.29        | 33.23        | 33.15        | 8.81        | 8.37        | 8.23        | 7.89        | 7.79        |
| + Ours   | <b>34.12</b> | <b>33.81</b> | <b>33.59</b> | <b>33.63</b> | <b>33.38</b> | <b>9.65</b> | <b>9.44</b> | <b>9.05</b> | <b>8.87</b> | <b>8.68</b> |

ing and LongLive, suggesting increased structural variation. For Self Forcing, DINOv3 diversity decreases slightly despite the improvement in CLIP diversity. Together with the consistent gains in visual quality, semantic alignment, distributional alignment, and human preference, these results suggest that Mask Forcing jointly improves generation quality and diversity.

## 6.5 INTERACTIVE VIDEO WORLD MODEL APPLICATION

We further explore Mask Forcing on autoregressive video generation with camera control. Current interactive world models extend video generation by conditioning future observations on user inputs (Team et al., 2026a;b; HunyuanWorld, 2025; Wang et al., 2026b; Zhao et al., 2026), such as actions, camera motions, or instructions. We evaluate Mask Forcing in a camera-controlled image-to-video setting, where an input image serves as the initial frame, and the model generates subsequent frames following a specified camera trajectory. We utilize Wan2.2-5B-TI2V as the base model and train the bidirectional model with open-source data, such as SpatialVID (Wang et al., 2026a), OmniWorld (Zhou et al., 2025), RealCam-Vid (Zheng et al., 2025), DL3DV (Ling et al., 2024), Sekai (Li et al., 2026b), and MiraData (Ju et al., 2024). Then we train an AR model using teacher forcing and adopt consistency distillation to obtain a few-step generator. Both bidirectional and autoregressive models are trained on 5s videos following SolarWM (Huang et al., 2026a). We compare Self Forcing and Mask Forcing for further causal student distillation to align with the teacher’s distribution with DMD. Fig. 10 presents qualitative comparisons on the Sekai-Game and Sekai-Walking test set. Videos generated by Self Forcing become darker and lose fine-grained details as the rollout proceeds. This issue can be attributed to the mode-seeking behavior of the reverse KL objective in DMD and error accumulation during self-rollout. Mask Forcing mitigates this degradation, producing scenes with richer details throughout the rollout. We view this experiment as an initial evaluation of our method in the camera-controlled image-to-video setting. To enable long and precise interactive video generation, future work includes training with long sequences and rollouts (Yang et al., 2025) and the context update strategy for temporal extrapolation (Zhuang et al., 2026).

## 6.6 ALGORITHM

We provide detailed pseudocode for training with Mask Forcing in Algorithm 1. Mask Forcing requires no additional forward passes, external data, or post-training stages and can be seamlessly integrated into existing self-rollout training pipelines. Mask Forcing constructs a dual-noise input at each denoising step by mixing the original- and lower-noise states with a random mask. The

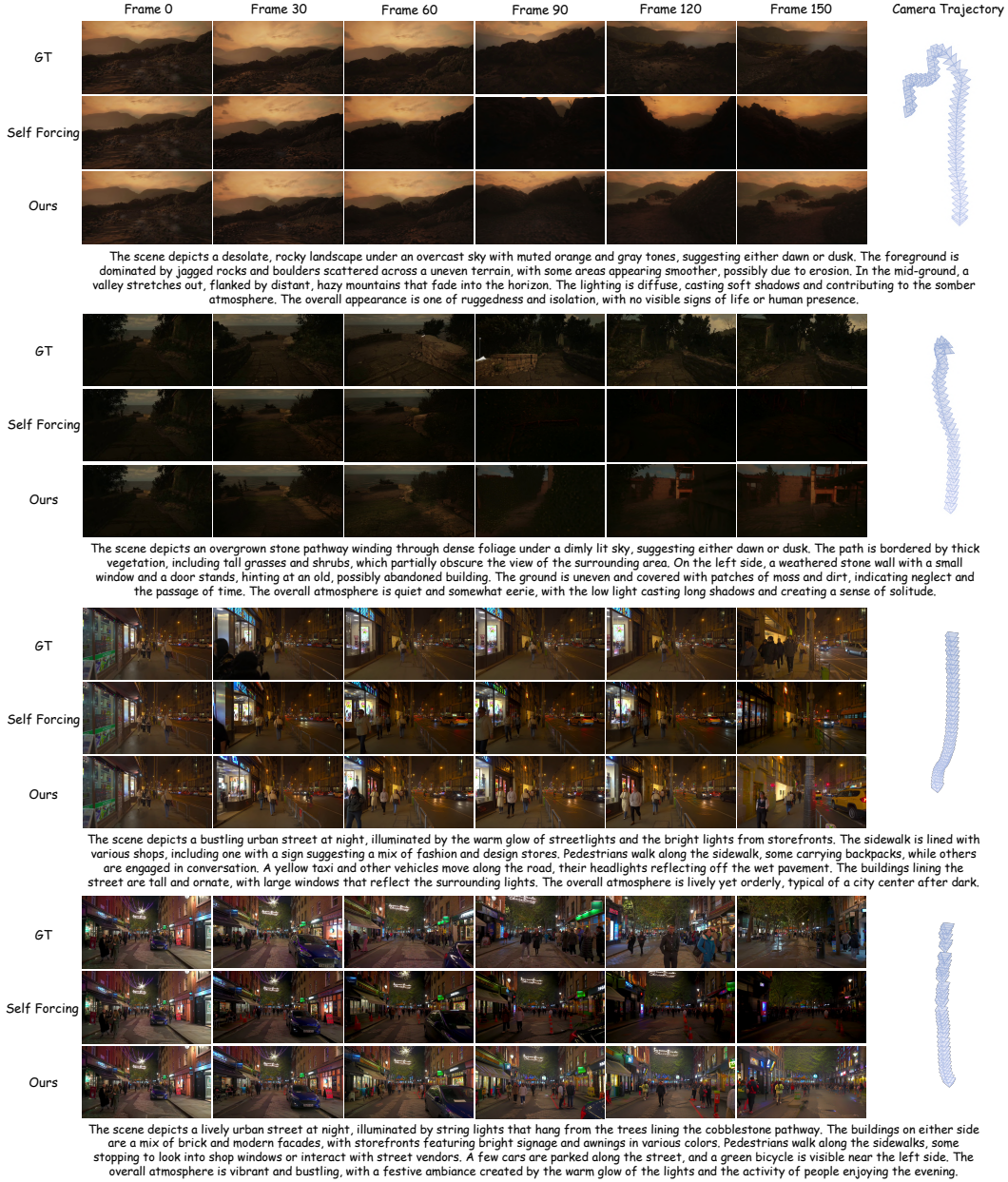


Figure 10: Qualitative comparison of Self Forcing and Mask Forcing (Ours) on Sekai-Game and Sekai-Walking. As the rollout progresses, Self Forcing videos darken and lose fine details. Mask Forcing mitigates this degradation, preserving rich details throughout generation.

perturbed input is then denoised at the original timestep, and the final clean prediction is cached as context for subsequent blocks.

## 6.7 MORE QUALITATIVE RESULTS

We present additional qualitative comparisons on single-prompt short video generation on the 100-prompt set and MovieGen in Figs. 11-13. Qualitative comparisons across different methods show that Mask Forcing consistently improves the visual quality of the generated videos, enhancing them with greater realism and richer detail while preventing over-saturated and over-smoothed artifacts.

---

**Algorithm 1** Mask Forcing Training

---

**Require:** Denoising timesteps  $\mathcal{T} = \{t_1, \dots, t_T\}$  with  $t_T = 1$  and clean endpoint  $t_0 = 0$   
**Require:** Text prompt  $c$ ; number of autoregressive chunks  $F$ , latent frames per chunk  $L$ , spatial latent resolution  $H \times W$   
**Require:** AR diffusion model  $G_\theta$  (returns KV embeddings via  $G_\theta^{\text{KV}}$ )  
**Require:** Mask ratio  $\alpha$ , timestep window size  $\Delta$  in training timestep units, number of training timesteps  $N_t$ , normalized timestep floor  $t_{\min}$   
**Notation:**  $\Psi(x, \varepsilon, t) = (1 - t)x + t\varepsilon$

- 1: **loop**
- 2:   Initialize KV cache  $\text{KV} \leftarrow []$  and model output  $X_\theta \leftarrow []$
- 3:   Sample exit index  $s \sim \text{Uniform}\{1, \dots, T\}$
- 4:   **for** chunk  $i = 1, \dots, F$  **do**
- 5:     Sample initial noisy latent  $x_i^{t_T} \sim \mathcal{N}(0, I)$
- 6:     Sample a per-frame binary mask  $M^i \in \{0, 1\}^{L \times H \times W}$  with ratio  $\alpha$
- 7:     **if**  $s = T$  **then**
- 8:       Enable gradient computation
- 9:     **else**
- 10:      Disable gradient computation
- 11:     **end if**
- 12:      $\hat{x}_{i,T}^0 \leftarrow G_\theta(x_i^{t_T}; c, t_T, \text{KV})$   $\triangleright$  initial unperturbed prediction
- 13:     **if**  $s < T$  **then**
- 14:       **for** denoising step  $j = T - 1, T - 2, \dots, s$  **do**
- 15:          $t'_k \sim \text{Uniform}([\max(t_{\min}, t_j - \Delta/N_t), t_j])$
- 16:         Sample  $\varepsilon_j \sim \mathcal{N}(0, I)$
- 17:          $x_i^{t'_k} \leftarrow \Psi(\hat{x}_{i,j+1}^0, \varepsilon_j, t'_k)$ ,    $x_i^{t_j} \leftarrow \Psi(\hat{x}_{i,j+1}^0, \varepsilon_j, t_j)$
- 18:          $x_i^{t_{\text{mix}}} \leftarrow M^i \odot x_i^{t'_k} + (1 - M^i) \odot x_i^{t_j}$
- 19:         **if**  $j = s$  **then**
- 20:           Enable gradient computation
- 21:         **else**
- 22:           Disable gradient computation
- 23:         **end if**
- 24:          $\hat{x}_{i,j}^0 \leftarrow G_\theta(x_i^{t_{\text{mix}}}; c, t_j, \text{KV})$   $\triangleright$  perturbed prediction at step  $j$
- 25:       **end for**
- 26:     **end if**
- 27:     Append  $\hat{x}_{i,s}^0$  to  $X_\theta$
- 28:     Disable gradient computation
- 29:      $\text{kv}^i \leftarrow G_\theta^{\text{KV}}(\hat{x}_{i,s}^0; c, t_0, \text{KV})$   $\triangleright$  cache the exit prediction
- 30:      $\text{KV.append}(\text{kv}^i)$
- 31:   **end for**
- 32:   Update  $\theta$  via the distribution matching loss  $\mathcal{L}_{\text{DMD}}(X_\theta)$
- 33: **end loop**

---



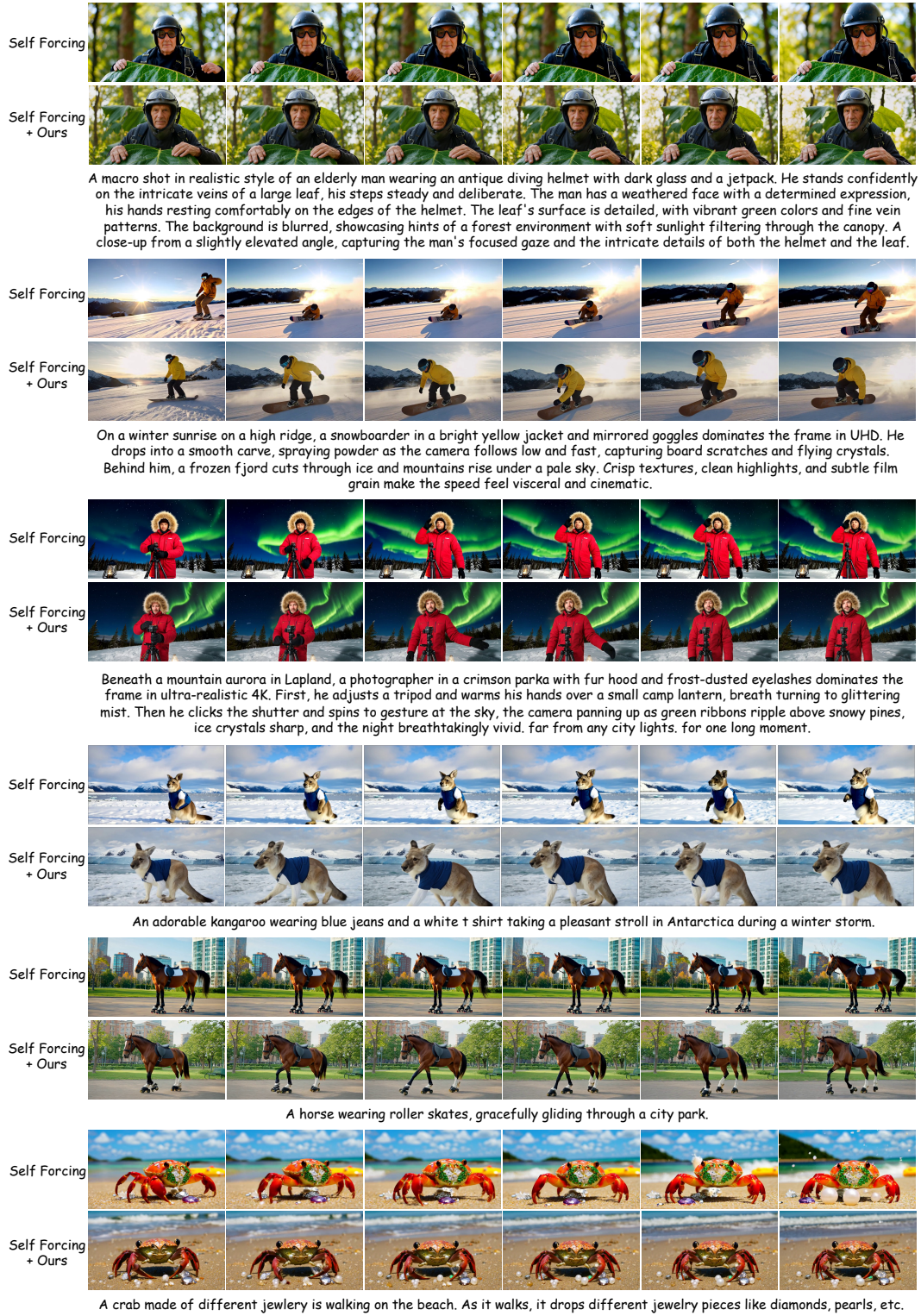


Figure 11: Additional qualitative comparisons between Self Forcing and Self Forcing (+Ours).





Figure 12: Additional qualitative comparisons between LongLive and LongLive (+Ours).



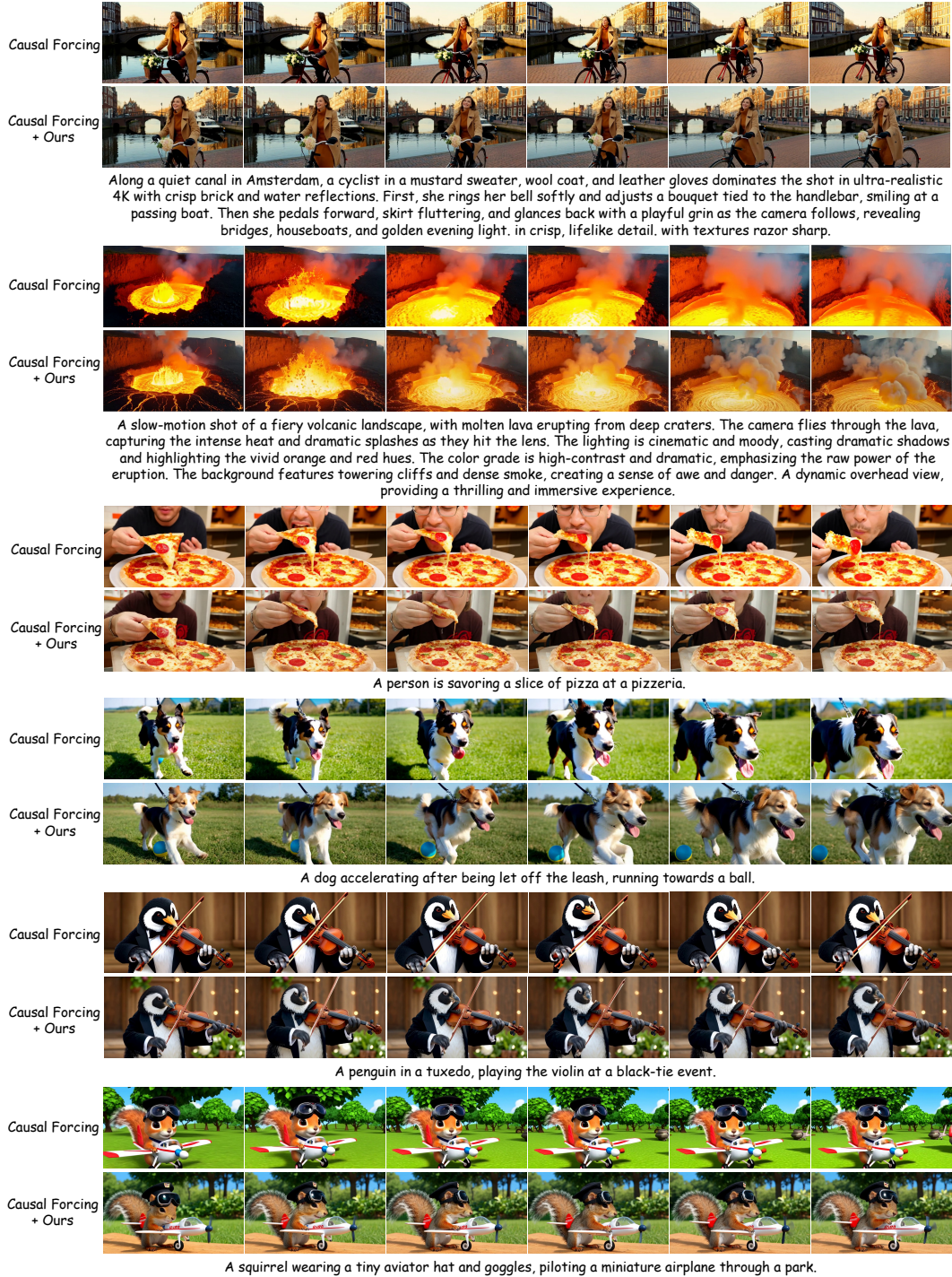


Figure 13: Additional qualitative comparisons between Causal Forcing and Causal Forcing (+Ours).