DYNPROMPT-DRIVEN: UNLEASHING MOTION DYNAMICS IN AUTOREGRESSIVE VIDEO GENERATION

Anonymous authors
Paper under double-blind review

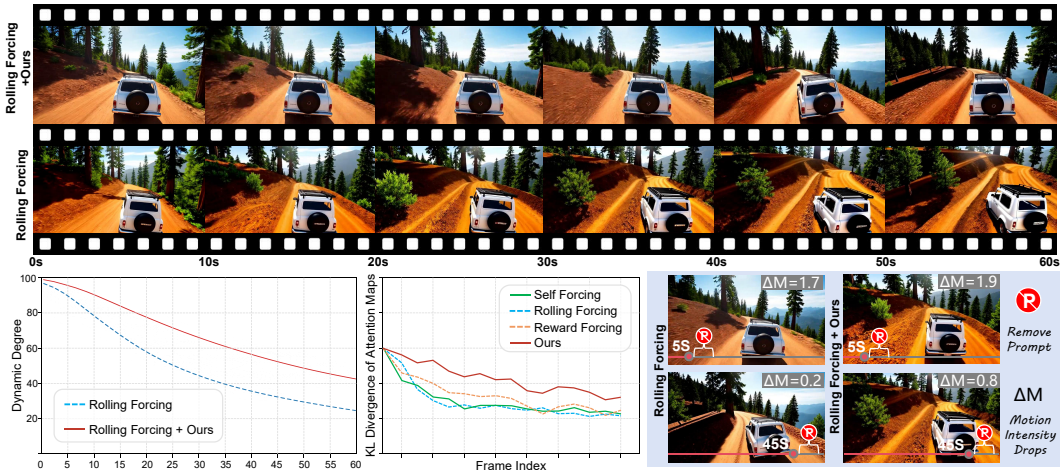

Figure 1: **Top:** our method substantially improves motion dynamics during 60-second rollouts with Rolling Forcing. **Bottom left:** the cumulative fraction of dynamic segments declines during rollout; our method mitigates this degradation. **Bottom middle:** KL divergence between consecutive text cross-attention maps decreases during rollout, with our method maintaining higher values. **Bottom right:** prompt removal has a smaller effect on the baseline's motion at later rollout times, indicating reduced sensitivity to continued text conditioning. Our method preserves greater sensitivity at later stages.

ABSTRACT

Autoregressive video generation has emerged as a promising paradigm for efficient streaming and video synthesis, and has witnessed rapid progress in recent years. However, existing autoregressive video diffusion models are prone to motion dynamics degradation during generation: in certain scenarios, videos become increasingly static as generation proceeds, and later segments tend to exhibit weaker motion than early ones. In this work, we ask a central question: *how can we explicitly unleash and preserve motion dynamics in autoregressive video generation, thereby enhancing overall video dynamism?* Cumulative dynamic-degree measurements, text cross-attention analysis, and paired prompt-removal experiments show weaker late-stage text contributions to motion in Rolling Forcing. Motivated by this observation, we propose **DynPrompt-Driven**, a dynamic-prompt-driven framework for autoregressive video diffusion. Specifically, we introduce **Temporal Prompt Recalibration (TPR)**, which dynamically recalibrates static prompt embeddings using the accumulated visual KV cache, enabling different textual tokens to be adaptively activated as time evolves, maintaining effective text guidance throughout long autoregressive rollouts. Furthermore, we design **Motion-Guided Gradient Alignment (MGGA)**, which leverages high-motion query videos to estimate motion-related gradient alignment of generated videos, steering distribution matching distillation (DMD) toward high-dynamism directions through motion-aware reweighting of individual training samples during optimization. Extensive experiments on video generation benchmarks demonstrate that our method substantially improves dynamic degree over strong autoregressive video generation baselines, with modest additional parameters and negligible inference overhead.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

1 INTRODUCTION

With the development of diffusion models, video generation has made remarkable progress, evolving from early video diffusion methods (Ho et al., 2022b; Blattmann et al., 2023; Singer et al., 2022; Ho et al., 2022a; Bar-Tal et al., 2024; Polyak et al., 2024) to high-quality latent video diffusion systems and large-scale video foundation models (Wan et al., 2025; Lin et al., 2024; Yang et al., 2024c; Kong et al., 2024; Chen et al., 2025). Recently, diffusion Transformer architectures have replaced convolutional U-Nets with Transformer denoisers, further improving model scalability and generation quality (Peebles & Xie, 2023; Ma et al., 2024; Chen et al., 2024; Esser et al., 2024). However, high-quality video generation remains computationally expensive, especially when synthesizing videos with dense temporal attention.

Autoregressive video generation reduces substantial computational overhead by generating videos segment by segment and reusing historical context through causal attention and KV caching. This paradigm has been widely explored in discrete token-based video generators, long-video diffusion methods, streaming generation systems, and recent causal video diffusion models (Yan et al., 2021; Ge et al., 2022; Yu et al., 2023; 2024; Harvey et al., 2022; Henschel et al., 2025; Yin et al., 2025; Sand.ai et al., 2025). Recently, Self Forcing and its follow-up works have shown that bidirectional video diffusion models can be distilled into autoregressive video synthesizers, thereby substantially narrowing the train–test gap during rollout (Huang et al., 2025; Cui et al., 2025; Yin et al., 2025; Yang et al., 2026; Liu et al., 2026). Reward Forcing further introduces reward-guided distribution matching to improve the efficiency and alignment of streaming video generation (Lu et al., 2026). These works demonstrate that autoregressive diffusion models can effectively enable fast inference and semantically aligned video generation.

Despite this progress, *long autoregressive rollouts remain susceptible to progressive motion degradation*. Our experimental analyses in Figure 1 show that, in some scenes, actions gradually slow down or nearly stop as generation proceeds. The cumulative fraction of dynamic segments also decreases with rollout duration. Although existing methods have improved temporal consistency, train–test alignment, inference efficiency, and long-range coherence (Huang et al., 2025; Cui et al., 2025; Yang et al., 2026; Liu et al., 2026), sustaining motion dynamics over long sequences remains an open challenge.

To investigate this degradation, we first conduct paired prompt-removal experiments to examine the contribution of text conditioning at different generation stages. We remove prompt conditioning at 5 or 45 seconds and compare the mean motion intensity over the following 5 seconds with the corresponding prompt-retained continuation. For Rolling Forcing, removal reduces motion intensity by 1.7 at 5 seconds but only 0.2 at 45 seconds, indicating a weaker contribution of continued text conditioning to late-stage motion. To further examine this change, we measure the KL divergence between consecutive text cross-attention maps. The divergence decreases during baseline rollouts, indicating increasingly similar attention maps and less temporal variation in text attention. Together with the prompt-removal results, this supports the observation that the prompt's influence on motion weakens as generation proceeds, a phenomenon we call **prompt suppression**. One possible explanation is that later motion becomes increasingly driven by historical visual information in the KV cache, while the static prompt provides insufficient ongoing guidance. This motivates adapting text conditioning to the evolving generation state.

To address this issue, we propose **DynPrompt-Driven**, which enhances motion dynamics through adaptive prompt conditioning and distillation optimization. **Temporal Prompt Recalibration (TPR)** dynamically recalibrates static prompt embeddings using the accumulated visual KV cache, allowing different textual semantic components to be adaptively activated throughout generation. **Motion-Guided Gradient Alignment (MGGA)** further strengthens motion during distribution matching distillation (DMD). We sample reference videos from a predefined high-motion video set, use their gradient directions as motion guidance, and reweight generated samples according to their gradient alignment with the references. This encourages distillation updates that improve the generator's motion dynamics.

Adapting our method to Rolling Forcing alleviates the slow-motion behavior in Figure 1, with a higher cumulative fraction of dynamic segments throughout the rollout. At 45 seconds, prompt removal reduces motion intensity by 0.8 with our method, compared with 0.2 for the baseline, indicating a stronger late-stage contribution of text conditioning. Our method also maintains higher

KL divergence between consecutive text cross-attention maps, indicating greater temporal variation in text attention. Together, these results support more sustained textual influence on motion and reduced motion degradation during long rollouts.

Our contributions are summarized as follows:

- Through motion statistics, paired prompt removal, and text cross-attention analysis, we examine the weakening influence of prompts on long-video motion and propose TPR to sustain text guidance by adapting prompt embeddings to visual history.
- We propose MGGA, which uses gradient directions from high-motion reference videos to guide DMD training and reweights generated samples according to gradient alignment to enhance motion dynamics.
- Experiments on short-video and 60-second long-video generation demonstrate improved dynamic degree across multiple autoregressive baselines while maintaining competitive visual quality and temporal consistency.

2 Related Work

Autoregressive video generation. Early token-based approaches such as VideoGPT, MAGVIT, and MAGVIT-v2 formulate video generation as discrete-token sequence modeling (Yan et al., 2021; Ge et al., 2022; Yu et al., 2023; 2024), while diffusion-based methods generate long videos via temporal windows or noise rescheduling (Harvey et al., 2022; Yin et al., 2023; Henschel et al., 2025; Lu et al., 2023; Qiu et al., 2023). More recently, CausVid converts bidirectional models into fast autoregressive diffusion (Yin et al., 2025); Self-Forcing and Self-Forcing++ bridge the train-test gap toward minute-scale generation (Huang et al., 2025; Cui et al., 2025); LONGLIVE and Rolling Forcing target real-time and rolling-window long-video synthesis (Yang et al., 2026; Liu et al., 2026). Stable Video Infinity learns to correct accumulated errors through error recycling (Li et al., 2026); MiMo strengthens history representations through masked history modeling (Luo et al., 2026); and Mixture of Contexts retrieves salient history through learned sparse attention routing (Cai et al., 2026). Our method follows the autoregressive distillation line but addresses a complementary issue: maintaining and amplifying motion dynamics during long-horizon rollout.

Diffusion distillation and acceleration. Diffusion distillation reduces sampling cost while preserving quality, with methods such as progressive distillation, consistency models, and latent consistency models achieving few-step generation via trajectory matching (Salimans & Ho, 2022; Song et al., 2023; Luo et al., 2023). Distribution Matching Distillation and its variants enable one-step generation by aligning distributions via score differences (Yin et al., 2024b;a), while Adversarial Diffusion Distillation combines adversarial training with distillation for fast synthesis (Sauer et al., 2023). Reward- and preference-based optimization further aligns diffusion models with task-specific objectives (Black et al., 2024; Fan et al., 2023; Xu et al., 2023; Lu et al., 2026). Our Motion-Guided Gradient Alignment differs from generic reward guidance by using motion-masked videos and gradient-direction alignment to dynamically reweight DMD training for motion enhancement.

Motion estimation and video evaluation. Learning-based optical-flow estimators such as RAFT estimate dense frame-to-frame motion (Teed & Deng, 2020), while point tracking methods including CoTracker, OmniMotion, and AllTracker enable robust long-range motion analysis (Karaev et al., 2024; Wang et al., 2023; Harley et al., 2025). Wu et al. (2026) use motion-aware gradient attribution to select offline fine-tuning subsets for improving video dynamics. For evaluation, FVD, FID, VBench, and EvalCrafter measure video realism, quality, temporal consistency, and generation capability (Unterthiner et al., 2019; Heusel et al., 2017; Huang et al., 2024a; Liu et al., 2024). In our experiments, we report per-dimension VBench metrics to demonstrate the effectiveness of the proposed method.

3 Method

3.1 Preliminaries

Autoregressive video diffusion models. In autoregressive video diffusion models, an N-frame sequence $\mathbf{x}_{1:N}$ follows $p(\mathbf{x}_{1:N}) = \prod_{i=1}^{N} p(\mathbf{x}_i \mid \mathbf{x}_{<i})$. Self-Forcing (Huang et al., 2025) introduces an autoregressive self-rollout mechanism that aligns training with inference: during training, each

frame $\mathbf{x}_i$ undergoes iterative denoising conditioned on previously *generated* (rather than ground-truth) frames, thereby bridging the train–test gap caused by exposure bias. A few-step diffusion model G_θ approximates each conditional $p(\mathbf{x}_i \mid \mathbf{x}_{<i})$. Given denoising timesteps $\{t_0, t_1, \ldots, t_K\}$, at step t_k the model processes the noisy frame $\mathbf{x}_i^{t_k}$ conditioned on $\mathbf{x}_{<i}$, then reintroduces controlled Gaussian noise via the forward process Ψ to yield $\mathbf{x}_i^{t_{k-1}}$ for the next step. A sample from the model's conditional distribution is generated as:

$$\mathbf{x}_i = f_{\theta,t_1} \circ f_{\theta,t_2} \circ \cdots \circ f_{\theta,t_K}\left(\mathbf{x}_i^{t_K}\right), \tag{1}$$

where $f_{\theta,t_k}(\mathbf{x}_i^{t_k}) = \Psi\big(G_\theta(\mathbf{x}_i^{t_k}, t_k, \mathbf{x}_{<i}), t_{k-1}\big)$ and $\mathbf{x}_i^{t_K} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$. This sampling procedure induces the conditional distribution $p_\theta(\mathbf{x}_i \mid \mathbf{x}_{<i})$. While models can output multi-frame chunks per step (Yin et al., 2025; Huang et al., 2025; Cui et al., 2025), we term each chunk a "frame" for simplicity. The text prompt $\mathbf{c}$ is encoded into a static embedding sequence $\mathbf{E} \in \mathbb{R}^{L\times d}$ via a T5 encoder (Raffel et al., 2020) and injected into G_θ through cross-attention at every generation step.

Distribution matching distillation. DMD (Yin et al., 2024b;a) distills a multi-step diffusion model into a few-step generator G_θ by minimizing the reverse KL divergence between the real distribution $p_{\text{real}}(\mathbf{x})$ and the generated distribution $p_{\text{fake}}(\mathbf{x})$ across timesteps:

$$\begin{aligned}\nabla_\theta \mathcal{L}_{\text{DMD}} &\triangleq \mathbb{E}_t[\nabla_\theta D_{\text{KL}}(p_{\text{fake},t}(\mathbf{x}_t) \,\|\, p_{\text{real},t}(\mathbf{x}_t))] \\ &\approx -\mathbb{E}_{t,\boldsymbol{\epsilon}}\left[\Big(s_{\text{real}}(\Psi(G_\theta(\boldsymbol{\epsilon}), t), t) - s_{\text{fake}}(\Psi(G_\theta(\boldsymbol{\epsilon}), t), t)\Big)\frac{dG_\theta(\boldsymbol{\epsilon})}{d\theta}\right],\end{aligned} \tag{2}$$

where $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ and Ψ denotes the forward diffusion process at timestep t. The score function is defined as:

$$s_{\text{real}}(\mathbf{x}_t, t) = \nabla_{\mathbf{x}_t} \log p_{\text{real},t}(\mathbf{x}_t) = -\frac{\mathbf{x}_t - \alpha_t\, \mu_{\text{real}}(\mathbf{x}_t, t)}{\sigma_t^2}, \tag{3}$$

where μ_{real} is the denoised estimate and α_t, σ_t are noise schedule parameters (Song et al., 2021a; Nichol & Dhariwal, 2021). DMD freezes the pretrained μ_{real} (teacher) and updates μ_{fake} (fake model) online on generator outputs. Since DMD training requires no ground-truth video data, Self-Forcing (Huang et al., 2025) conditions each frame on previously self-generated histories during training, effectively mitigating exposure bias in autoregressive rollout.

3.2 Temporal Prompt Recalibration

Observation. Long autoregressive rollouts can preserve recognizable scene content while progressively losing motion. In Figure 1, the car-driving example develops slow-motion behavior, and the cumulative fraction of dynamic segments decreases as generation proceeds. This motivates examining whether the original prompt continues to influence motion after the scene has been established. Paired prompt-removal experiments reveal a temporal difference in this influence. For Rolling Forcing, removing the prompt at 5 seconds reduces the mean motion intensity over the following 5 seconds by 1.7 relative to the corresponding prompt-retained continuation; removal at 45 seconds reduces it by only 0.2. The later continuation is therefore less sensitive to text conditioning in terms of measured motion intensity. Our adaptation retains a larger reduction of 0.8 at 45 seconds, suggesting that this weakening can be alleviated. Concurrently, the KL divergence between consecutive text cross-attention maps decreases during baseline rollouts, indicating less temporal variation in the attention distribution. Attention stability alone does not establish prompt suppression, but, together with the removal experiment, it motivates maintaining effective text guidance as the visual history evolves. The design goal is sustained prompt influence on motion; attention variation provides supporting evidence rather than a separate optimization objective.

Temporal Prompt Recalibration (TPR). To counteract prompt suppression, we introduce Temporal Prompt Recalibration (TPR). A static prompt supplies the same embedding sequence throughout generation, even as object poses, spatial relationships, and camera views change. The denoiser's attention can already vary with its visual queries; Temporal Prompt Recalibration adds an explicit feedback path from the generated visual history to the *text embedding itself*. This allows textual emphasis to adapt to the evolving generation state.

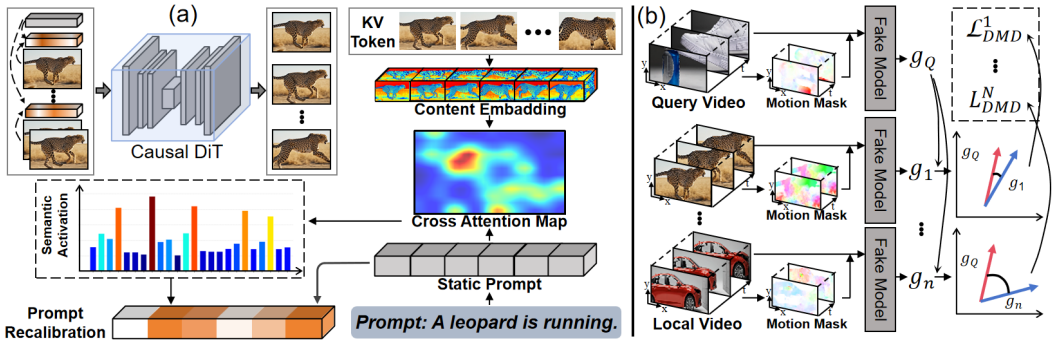

Figure 2: **Overview of DynPrompt-Driven. (a) Temporal Prompt Recalibration.** The static prompt embedding is recalibrated at each autoregressive step by computing cross-attention between text tokens and the accumulated KV-cache, dynamically adjusting the activation of different semantic components so that the prompt embedding evolves continuously throughout generation. **(b) Motion-Guided Gradient Alignment.** Motion cues are extracted from both the query and local videos and fed into the fake model to obtain parameter gradient directions. The cosine similarity between each local gradient and the query gradient is used to reweight per-sample contributions to the distribution matching distillation loss.

The prompt contains multiple semantic components, such as appearance, scene layout, and actions. Their relative influence may need to change as the scene develops: motion-related content should continue to guide generation even when much of the scene's appearance is already established. The visual KV cache provides an available representation of this generated context. Temporal Prompt Recalibration uses the correspondence between each text token and the cached visual features to predict a token-specific recalibration weight. The resulting embedding retains the original token sequence while adapting the relative strength of its components. Compared with fixed-interval prompt switching, this provides context-dependent adjustment without manually constructing a sequence of replacement prompts.

Concretely, let $\mathbf{E}$ denote the static embedding containing L text tokens. At generation step t, let $\mathcal{K}^{(t)} = \{(\mathbf{k}_i^{(t)}, \mathbf{v}_i^{(t)})\}_{i=1}^{N_t}$ denote the available visual cache from previously generated chunks, and let $\mathbf{K}^{(t)} \in \mathbb{R}^{N_t \times d'}$ stack its key vectors. As shown in Figure 2(a), Temporal Prompt Recalibration computes a text-to-history attention map $\mathbf{A}^{(t)} \in \mathbb{R}^{L \times N_t}$. Feature projections into the shared d'-dimensional attention space are implicit in the text–cache similarity computation below. Each row $\mathbf{A}^{(t)}_{j,:}$ is mapped to a scalar weight $m_j^{(t)} \in [0, 1]$ through a learned linear projection followed by a sigmoid, forming the recalibration mask $\mathbf{M}^{(t)} \in \mathbb{R}^L$:

$$\mathbf{A}^{(t)} = \mathrm{softmax}\left(\frac{\mathbf{E}\,\mathbf{K}^{(t)\top}}{\sqrt{d'}}\right), \qquad \tilde{\mathbf{E}}^{(t)} = \mathbf{M}^{(t)} \odot \mathbf{E}, \tag{4}$$

where $\odot$ denotes element-wise multiplication broadcast over the embedding dimension. Each attention row describes a token's correspondence with the visual history, and the learned mapping converts this context profile into a conditioning strength. Scaling an entire token vector preserves its embedding direction while changing its magnitude relative to other tokens. Thus, Temporal Prompt Recalibration can adjust the relative emphasis on motion-related content while retaining the original textual representations.

The recalibrated embedding $\tilde{\mathbf{E}}^{(t)}$ replaces the static embedding in the DiT's text cross-attention layers to condition the current chunk. Once that chunk is generated, its visual features are appended to the cache for the next step:

$$\mathbf{x}^{(t)} \sim p_\theta\left(\mathbf{x}^{(t)} \,\middle|\, \mathcal{K}^{(t)}, \tilde{\mathbf{E}}^{(t)}\right), \qquad \mathcal{K}^{(t+1)} = \mathcal{K}^{(t)} \cup \left\{\left(\mathbf{k}^{(t)}, \mathbf{v}^{(t)}\right)\right\}. \tag{5}$$

This forms a feedback loop: the visual history determines the text recalibration, the recalibrated prompt guides the next chunk, and the new chunk updates the context for subsequent recalibration. The same cache therefore supports both historical visual conditioning and adaptation of the text condition.

The Temporal Prompt Recalibration parameters are jointly trained with the causal DiT during DMD distillation, allowing the recalibration weights to be learned through their effect on generation. When combined with Motion-Guided Gradient Alignment, sample reweighting further favors training updates aligned with high-dynamic reference videos, linking contextual prompt adaptation to the goal of motion enhancement. At inference, Temporal Prompt Recalibration reuses the existing visual cache and requires no additional denoising pass through the generator.

3.3 Motion-Guided Gradient Alignment

To further enable TPR to be trained efficiently and to attend as much as possible to dynamic semantics, we propose Motion-Guided Gradient Alignment (MGGA).

High-dynamic query video library. We first curate a query set $\mathcal{V}_{\text{Q}}$ of high-dynamic videos generated by Veo 3 (Google DeepMind, 2025; Wiedemer et al., 2025). These videos cover common primitive motion modalities with prompts favoring simple backgrounds to reduce interference with motion semantic extraction. We use Veo 3 to synthesize these controlled high-dynamic samples. Each of the 10 motion categories contains 10 videos, yielding $|\mathcal{V}_{\text{Q}}| = 100$ in total. The motion categories are designed to span a diverse range of dynamics, including rigid body motion, articulated motion, fluid dynamics, and camera-induced motion, ensuring that the query gradients capture a broad and representative notion of high-dynamic content rather than being biased toward any single motion type.

Motion mask extraction. As shown in Figure 2(b), given a video $\mathbf{v}$, we use AllTracker (Harley et al., 2025) to extract per-frame motion information, including: (i) inter-frame optical flow $\mathbf{F} \in \mathbb{R}^{H\times W\times 2}$, (ii) motion magnitude $\mathbf{S} \in \mathbb{R}^{H\times W}$, and (iii) motion confidence $\mathbf{C} \in \mathbb{R}^{H\times W}$. We fuse these signals into a soft motion mask:

$$\mathbf{M}_{\text{motion}}(\mathbf{v}) = \sigma(\alpha \cdot \mathbf{S} \cdot \mathbf{C} - \beta)\,, \tag{6}$$

where α and β are threshold hyperparameters. The motion confidence $\mathbf{C}$ plays a critical role here: it down-weights regions where the optical flow estimate is unreliable, such as occluded areas or textureless surfaces, preventing noisy flow estimates from corrupting the mask. The masked video $\hat{\mathbf{v}} = \mathbf{M}_{\text{motion}}(\mathbf{v}) \odot \mathbf{v}$ retains only the dynamic foreground regions, suppressing static background content that would otherwise introduce spurious correlations into the gradient alignment.

Gradient direction computation. At each training iteration, we sample a query video $\mathbf{v}_{\text{Q}} \sim \mathcal{V}_{\text{Q}}$ and compute its masked version $\hat{\mathbf{v}}_{\text{Q}}$. Let $\mathcal{B} = \{\mathbf{v}_1, \ldots, \mathbf{v}_B\}$ denote the batch of locally generated videos across all GPUs, with corresponding masked versions $\{\hat{\mathbf{v}}_i\}$. We pass both the query and local masked videos through the fake model ϕ to obtain their respective gradient directions with respect to parameters of ϕ:

$$\mathbf{g}_{\text{Q}} = \nabla_\phi \mathcal{L}_{\text{fake}}(\hat{\mathbf{v}}_{\text{Q}};\, \phi), \qquad \mathbf{g}_i = \nabla_\phi \mathcal{L}_{\text{fake}}(\hat{\mathbf{v}}_i;\, \phi). \tag{7}$$

Here $i = 1, \ldots, B$ indexes the generated samples in the batch. By operating in gradient space rather than feature space, MGGA measures alignment at the level of how each sample drives the model's parameter updates, providing a proxy for whether a sample's update direction aligns with that of a high-motion reference. Note that $\mathbf{g}_{\text{Q}}$ is used solely to compute alignment weights and is *not* used to update ϕ, ensuring that the query videos do not directly alter the fake model but only serve as a directional anchor for reweighting.

Motion-modulated DMD loss. We compute the cosine similarity between each local gradient and the query gradient to measure how well the generated video aligns with the high-dynamic direction. The resulting similarity scores are converted into batch-mean normalized importance weights:

$$\bar{w}_i = B \cdot \frac{\exp(\tau \cdot \cos(\mathbf{g}_i,\, \mathbf{g}_{\text{Q}}))}{\sum_{j=1}^{B} \exp(\tau \cdot \cos(\mathbf{g}_j,\, \mathbf{g}_{\text{Q}}))}, \quad \cos(\mathbf{g}_i,\, \mathbf{g}_{\text{Q}}) = \frac{\mathbf{g}_i \cdot \mathbf{g}_{\text{Q}}}{\|\mathbf{g}_i\|\, \|\mathbf{g}_{\text{Q}}\|}, \tag{8}$$

where $B = |\mathcal{B}|$ is the batch size and τ is an inverse temperature, set to 1 by default. The factor B rescales the softmax weights so that their batch mean equals one, thereby preserving the overall scale

of the original per-sample DMD objective. We then use $\{\bar{w}_i\}$ as importance weights to modulate the per-sample DMD loss:

$$\mathcal{L}_{\text{MGGA}}(\theta) = \frac{1}{B}\sum_{i=1}^{B} \bar{w}_i \cdot \mathcal{L}_{\text{DMD}}(\mathbf{v}_i;\, \theta). \tag{9}$$

This formulation amplifies the contribution of samples with higher gradient alignment and suppresses that of samples with lower alignment, promoting the motion-aware learning capacity of Temporal Prompt Recalibration and steering the generator toward high-dynamic outputs. Crucially, MGGA introduces no additional learnable parameters and requires an additional fake-model gradient-projection step per training iteration. In practice, we compute projected gradient directions using a structured projection based on Fastfood (Le et al., 2013), as described in Appendix C, avoiding explicit storage of full parameter gradients.

4 Experiments

We evaluate short- and long-video generation, conduct component ablations, and conclude with a user study. Implementation and short-video evaluation details are given in Appendix A; the long-video protocol and visual comparisons are provided in Appendices F and G.

Table 1: **Video performance comparison with baselines.** We compare our method with representative open-source video generation models of comparable scale. Best VBench scores in **bold**, second-best underlined.

Model	#Params	Throughput (FPS) ↑	VBench evaluation scores ↑				
			Total Score	Quality Score	Semantic Score	Dynamic Degree	Motion Smoothness
Diffusion models							
Wan2.1	1.3B	0.78	<u>84.26</u>	<u>85.30</u>	80.09	<u>73.61</u>	97.97
Autoregressive models							
SkyReels-V2	1.3B	0.49	82.67	84.70	74.53	68.77	97.57
CausVid	1.3B	17.0	82.88	83.93	78.69	61.11	97.98
Self Forcing	1.3B	17.0	83.80	84.59	80.64	66.38	98.24
LongLive	1.3B	20.7	83.22	83.68	**81.37**	40.83	**98.79**
Rolling Forcing	1.3B	15.8	81.22	84.08	69.78	68.05	98.57
Reward Forcing	1.3B	23.1	84.13	84.84	<u>81.32</u>	68.05	98.32
Ours	1.3B	17.1	**84.60**	**85.62**	80.53	**83.88**	<u>98.64</u>

4.1 Short Video Generation

We comprehensively benchmark our method against a wide range of existing open-source video generation models, including Wan2.1 (Wan et al., 2025), SkyReels-V2 (Chen et al., 2025), CausVid (Yin et al., 2025), Self Forcing (Huang et al., 2025), LongLive (Yang et al., 2026), Rolling Forcing (Liu et al., 2026), and Reward Forcing (Lu et al., 2026). The overall VBench score is composed of both quality and semantic components, providing a holistic assessment of the generated videos. As shown in Table 1, our method effectively enhances the dynamics of generated video clips, improving the dynamic degree metric by at least 10.27 points over all baselines, which highlights the superior motion richness of our approach. Meanwhile, our method achieves a competitive overall score of 84.60, demonstrating that it can substantially boost video dynamics while simultaneously maintaining stable video quality and strong semantic fidelity. Notably, the introduction of the Temporal Prompt Recalibration module incurs only a marginal overhead of 47M additional parameters, with negligible impact on the video generation frames-per-second (FPS), making it highly practical for real-world deployment.

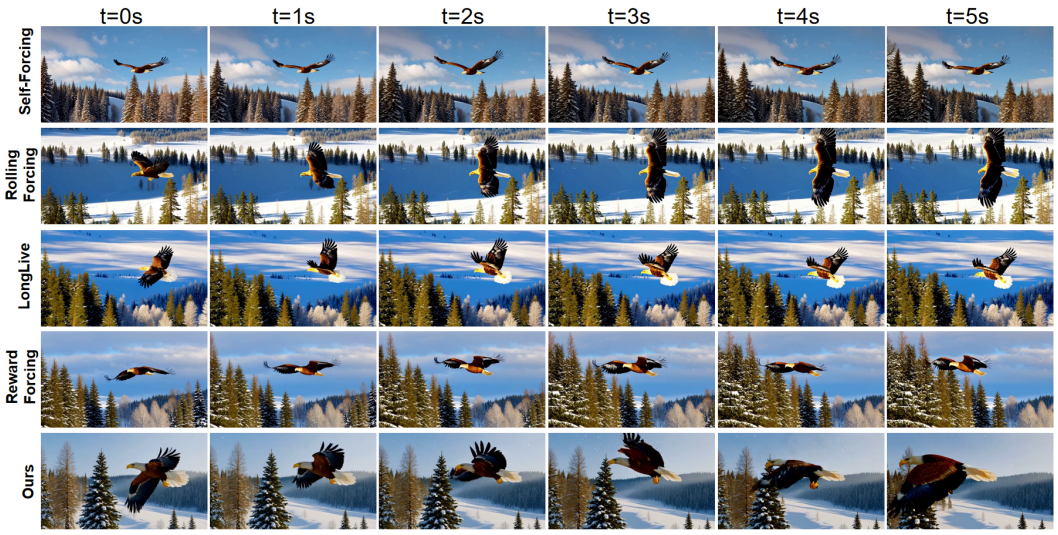

Figure 3: **Qualitative comparison**. Compared to existing open-source methods, our approach generates videos with significantly enhanced motion dynamics, while maintaining temporal consistency and overall visual quality.

Table 2: **60-second video generation comparison.** Each baseline is compared with its adaptation using our method. Higher is better for the six VBench-Long dimensions; lower is better for Drift.

Model	VBench-Long evaluation scores ↑						Drift ↓
	Subject Consistency	Background Consistency	Motion Smoothness	Dynamic Degree	Aesthetic Quality	Imaging Quality	
Rolling Forcing	98.23	96.66	98.94	27.00	61.12	71.34	2.22
Rolling Forcing + Ours	97.98	96.27	98.76	44.35	62.47	70.98	2.15
Reward Forcing	97.09	96.13	98.51	62.71	58.07	69.88	2.49
Reward Forcing + Ours	97.31	96.45	98.37	73.57	59.03	69.71	2.41

4.2 Long Video Generation

Table 2 compares Rolling Forcing and Reward Forcing with and without our method on 60-second video generation using six VBench-Long metrics and Drift. Dynamic degree increases from 27.00 to 44.35 (+17.35 points) and from 62.71 to 73.57 (+10.86 points), respectively. Both adaptations improve aesthetic quality and reduce Drift, indicating more stable imaging quality over time, while motion smoothness and imaging quality decrease slightly. Evaluation details and qualitative comparisons are provided in Appendices F and G.

The motion diagnostics in Figure 1 show a higher cumulative dynamic fraction for our Rolling Forcing adaptation. At 45 seconds, the paired prompt-removal effect on motion is 0.8 with our method versus 0.2 for Rolling Forcing, compared with 1.9 and 1.7 at 5 seconds, respectively. These observations support mitigation of weakening text contributions to motion.

4.3 Ablation Study

Impact of TPR and MGGA. We conduct ablation studies to validate the contribution of each proposed component: Temporal Prompt Recalibration (TPR) and Motion-Guided Gradient Alignment (MGGA). As shown in Table 3a, removing both modules reduces the model to the standard autoregressive distillation baseline, which relies on static text embeddings and yields the lowest dynamic degree of 66.38. Introducing TPR alone substantially improves the dynamic degree by 7.23 points, supporting the benefit of adaptive prompt conditioning for motion enhancement. Adding MGGA

Table 3: **Ablation studies. Left:** impact of each proposed module (TPR and MGGA). **Right:** comparison of prompt-conditioning configurations.

(a) Module ablation.

TPR	MGGA	VBench Scores ↑			
		Total	Quality	Semantic	Dynamic
Module Ablation					
×	×	83.80	84.59	80.64	66.38
✓	×	84.09	84.88	80.91	73.61
×	✓	83.92	84.71	80.77	71.28
✓	✓	84.60	85.62	80.53	83.88

(b) Prompt conditioning.

Strategy	VBench Scores ↑			
	Total	Quality	Semantic	Dynamic
Prompt Strategy				
Static	83.80	84.59	80.64	66.38
Fixed	82.15	82.69	80.01	71.28
Dynamic(Ours)	84.60	85.62	80.53	83.88

alone also yields a notable improvement in dynamic degree, demonstrating that motion-guided gradient reweighting provides an effective training signal toward high-dynamism generation. Combining both modules achieves the best dynamic degree of 83.88, with competitive total score and semantic score, indicating that TPR and MGGA are complementary: TPR improves prompt-driven motion activation at inference time, while MGGA steers the distillation process toward motion-rich outputs during training. When MGGA is disabled, all generated samples receive equal weight in the DMD objective; these configurations therefore serve as the uniform-weighting baseline. The 60-second component ablation is provided in Appendix F.1.

Comparison of prompt-conditioning configurations. Table 3b compares configurations using static prompts, fixed-interval switching, and dynamic prompt recalibration; the conditioning protocol is specified in Appendix A. The static configuration has the lowest dynamic degree (66.38). Fixed-interval switching reaches 71.28, with lower total and quality scores than our dynamic configuration, which achieves a dynamic degree of 83.88. These results compare the reported configurations; they do not by themselves isolate the effect of prompt conditioning from that of MGGA.

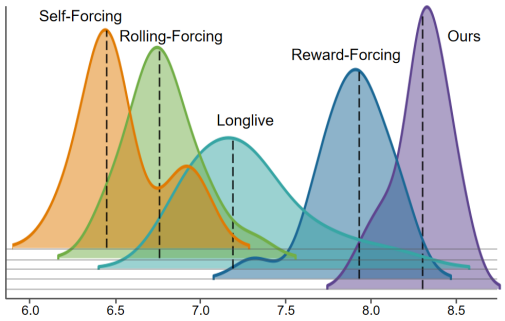

Figure 4: User study results for five methods. Participants rated motion richness and visual quality; the protocol is in Appendix E.

4.4 User Study

We conduct a user study with 20 adult participants to compare motion richness and visual quality across methods (Figure 4). The study protocol, including randomized presentation and rating criteria, is provided in Appendix E.

5 Conclusion

We presented DynPrompt-Driven, a dynamic-prompt-driven framework for improving motion dynamics in autoregressive video generation. Motivated by long-horizon motion degradation and paired evidence of weaker late-stage text contributions to motion, we introduced Temporal Prompt Recalibration to dynamically recalibrate prompt embeddings based on visual cache information. This adaptation allows textual guidance to evolve with the generated context throughout the autoregressive rollout. To further strengthen motion during distillation, we proposed Motion-Guided Gradient Alignment, which uses the alignment between fake-model gradients induced by generated and high-motion reference videos to reweight per-sample DMD losses. TPR operates during generation, while MGGA shapes training updates without introducing additional learnable model parameters. Built upon a self-forcing-style bidirectional-to-causal distillation pipeline, our method is designed to improve dynamic degree, motion persistence, and prompt-motion alignment while maintaining high visual fidelity. Experiments on autoregressive video generation demonstrate that DynPrompt-Driven effectively unleashes motion dynamics in autoregressive video diffusion.

REPRODUCIBILITY STATEMENT

The main paper specifies the model components, objectives, datasets, evaluation metrics, and inference costs. Detailed optimization settings, training procedures, evaluation protocols, and additional ablations are provided in the supplementary material to support reproduction of the reported results. Anonymous code can be provided to reviewers upon request during the discussion phase.

ETHICS STATEMENT

The user study involved 20 adult participants and was conducted solely for subjective evaluation of video motion richness and visual quality. All participants were informed of the purpose and procedure of the study and participated voluntarily. No personally identifiable or sensitive information was collected, and participants received no monetary or other compensation. Under our institution's policy, this anonymous, non-medical, low-risk video-rating study does not require formal ethics-board approval.

AI USE STATEMENT

We used Veo 3 to synthesize the 100 high-dynamic reference videos used by MGGA, as described in Section 3.3 and Appendix C. We used Qwen2-72B-Instruct to extend the filtered VidProM training prompts. Qwen2.5-7B-Instruct was used to rewrite VBench evaluation prompts, decompose prompts for the fixed-interval switching ablation, and classify prompt motion semantics in the additional gradient-alignment analysis (Section 4 and Appendix C). This study does not use additional external real-world video datasets for generator training. MGGA uses a small synthetic high-dynamic reference library generated from manually designed prompts, which serves only to compute gradient-alignment weights rather than as direct supervised targets. All text prompts leveraged in self-supervised training, generated from Qwen2-72B-Instruct (Yang et al., 2024a), are clean, safe, and intended solely for academic research purposes.

Generative AI tools were used to assist with manuscript language editing, organization, literature/reference checking, and consistency checking between the manuscript and implementation. No measured experimental data or quantitative results were generated by these manuscript-assistance tools. All AI-assisted content was reviewed and verified by the authors, who take responsibility for the final manuscript.

REFERENCES

Omer Bar-Tal, Hila Chefer, Omer Tov, Charles Herrmann, Roni Paiss, Shiran Zada, Ariel Ephrat, Junhwa Hur, Yuanzhen Li, Tomer Michaeli, et al. Lumiere: A space-time diffusion model for video generation. *arXiv preprint arXiv:2401.12945*, 2024.

Kevin Black, Michael Janner, Yilun Du, Ilya Kostrikov, and Sergey Levine. Training diffusion models with reinforcement learning. In *International Conference on Learning Representations*, 2024.

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.

Shengqu Cai, Ceyuan Yang, Lvmin Zhang, Yuwei Guo, Junfei Xiao, Ziyan Yang, Yinghao Xu, Zhenheng Yang, Alan Yuille, Leonidas Guibas, Maneesh Agrawala, Lu Jiang, and Gordon Wetzstein. Mixture of contexts for long video generation. In *International Conference on Learning Representations*, 2026.

Guibin Chen, Dixuan Lin, Jiangping Yang, Chunze Lin, Junchen Zhu, Mingyuan Fan, Hao Zhang, Sheng Chen, Zheng Chen, Chengcheng Ma, et al. SkyReels-V2: Infinite-length film generative model. *arXiv preprint arXiv:2504.13074*, 2025.

Junsong Chen, Jincheng Yu, Chongjian Ge, Lewei Yao, Enze Xie, Yue Wu, Zhongdao Wang, James Kwok, Ping Luo, Huchuan Lu, et al. PixArt-α: Fast training of diffusion transformer for photorealistic text-to-image synthesis. In *International Conference on Learning Representations*, 2024.

Justin Cui, Jie Wu, Ming Li, Tao Yang, Xiaojie Li, Rui Wang, Andrew Bai, Yuanhao Ban, and Cho-Jui Hsieh. Self-Forcing++: Towards minute-scale high-quality video generation. *arXiv preprint arXiv:2510.02283*, 2025.

Prafulla Dhariwal and Alexander Nichol. Diffusion models beat GANs on image synthesis. *Advances in Neural Information Processing Systems*, 34:8780–8794, 2021.

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 *International Conference on Machine Learning*, volume 235, pp. 12606–12633. PMLR, 2024.

Ying Fan, Olivia Watkins, Yuqing Du, Hao Liu, Moonkyung Ryu, Craig Boutilier, Pieter Abbeel, Mohammad Ghavamzadeh, Kangwook Lee, and Kimin Lee. DPOK: Reinforcement learning for fine-tuning text-to-image diffusion models. *Advances in Neural Information Processing Systems*, volume 36, 2023.

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 *European Conference on Computer Vision*, pp. 102–118, 2022.

Google DeepMind. Veo 3 model card. Technical report, 2025. Updated January 13, 2026. URL `https://storage.googleapis.com/deepmind-media/Model-Cards/Veo-3-Model-Card.pdf`.

Yuwei Guo, Ceyuan Yang, Anyi Rao, Maneesh Agrawala, Dahua Lin, and Bo Dai. AnimateDiff: Animate your personalized text-to-image diffusion models without specific tuning. In *International Conference on Learning Representations*, 2024.

Adam W Harley, Yang You, Xinglong Sun, Yang Zheng, Nikhil Raghuraman, Yunqi Gu, Sheldon Liang, Wen-Hsuan Chu, Achal Dave, Suya You, et al. AllTracker: Efficient dense point tracking at high resolution. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 5253–5262, 2025.

William Harvey, Saeid Naderiparizi, Vaden Masrani, Christian Weilbach, and Frank Wood. Flexible diffusion modeling of long videos. *Advances in Neural Information Processing Systems*, 35: 27953–27965, 2022.

Roberto Henschel, Levon Khachatryan, Hayk Poghosyan, Daniil Hayrapetyan, Vahram Tadevosyan, Zhangyang Wang, Shant Navasardyan, and Humphrey Shi. StreamingT2V: Consistent, dynamic, and extendable long video generation from text. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 2568–2577, 2025.

Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. GANs trained by a two time-scale update rule converge to a local Nash equilibrium. In *Advances in Neural Information Processing Systems*, volume 30, 2017.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems*, volume 33, 2020.

Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022.

Jonathan Ho, William Chan, Chitwan Saharia, Jay Whang, Ruiqi Gao, Alexey Gritsenko, Diederik P Kingma, Ben Poole, Mohammad Norouzi, David J Fleet, et al. Imagen video: High definition video generation with diffusion models. *arXiv preprint arXiv:2210.02303*, 2022a.

Jonathan Ho, Tim Salimans, Alexey Gritsenko, William Chan, Mohammad Norouzi, and David J Fleet. Video diffusion models. *Advances in Neural Information Processing Systems*, 35:8633–8646, 2022b.

Xun Huang, Zhengqi Li, Guande He, Mingyuan Zhou, and Eli Shechtman. Self Forcing: Bridging the train-test gap in autoregressive video diffusion. In *Advances in Neural Information Processing Systems*, volume 38, 2025.

Ziqi Huang, Yinan He, Jiashuo Yu, Fan Zhang, Chenyang Si, Yuming Jiang, Yuanhan Zhang, Tianxing Wu, Qingyang Jin, Nattapol Chanpaisit, Yaohui Wang, Xinyuan Chen, Limin Wang, Dahua Lin, Yu Qiao, and Ziwei Liu. VBench: Comprehensive benchmark suite for video generative models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024a.

Ziqi Huang, Fan Zhang, Xiaojie Xu, Yinan He, Jiashuo Yu, Ziyue Dong, Qianli Ma, Nattapol Chanpaisit, Chenyang Si, Yuming Jiang, Yaohui Wang, Xinyuan Chen, Ying-Cong Chen, Limin Wang, Dahua Lin, Yu Qiao, and Ziwei Liu. VBench++: Comprehensive and versatile benchmark suite for video generative models. *arXiv preprint arXiv:2411.13503*, 2024b.

Nikita Karaev, Ignacio Rocco, Benjamin Graham, Natalia Neverova, Andrea Vedaldi, and Christian Rupprecht. CoTracker: It is better to track together. In *European Conference on Computer Vision*, 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.

Quoc Le, Tamás Sarlós, and Alexander Smola. Fastfood – computing Hilbert space expansions in loglinear time. In *International Conference on Machine Learning*, volume 28(3), pp. 244–252. PMLR, 2013.

Wuyang Li, Wentao Pan, Po-Chien Luan, Yang Gao, and Alexandre Alahi. Stable video infinity: Infinite-length video generation with error recycling. In *International Conference on Learning Representations*, 2026.

Bin Lin, Yunyang Ge, Xinhua Cheng, Zongjian Li, Bin Zhu, Shaodong Wang, Xianyi He, Yang Ye, Shenghai Yuan, Liuhan Chen, et al. Open-Sora Plan: Open-source large video generation model. *arXiv preprint arXiv:2412.00131*, 2024.

Yaron Lipman, Ricky TQ Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. *arXiv preprint arXiv:2210.02747*, 2022.

Kunhao Liu, Wenbo Hu, Jiale Xu, Ying Shan, and Shijian Lu. Rolling Forcing: Autoregressive long video diffusion in real time. In *International Conference on Learning Representations*, 2026.

Xingchao Liu, Chengyue Gong, and Qiang Liu. Flow straight and fast: Learning to generate and transfer data with rectified flow. *arXiv preprint arXiv:2209.03003*, 2022.

Yaofang Liu, Xiaodong Cun, Xuebo Liu, Xintao Wang, Yong Zhang, Haoxin Chen, Yang Liu, Tieyong Zeng, Raymond Chan, and Ying Shan. EvalCrafter: Benchmarking and evaluating large video generation models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024.

Haoyu Lu, Guoxing Yang, Nanyi Fei, Yuqi Huo, Zhiwu Lu, Ping Luo, and Mingyu Ding. VDT: General-purpose video diffusion transformers via mask modeling. *arXiv preprint arXiv:2305.13311*, 2023.

Yunhong Lu, Yanhong Zeng, Haobo Li, Hao Ouyang, Qiuyu Wang, Ka Leong Cheng, Jiapeng Zhu, Hengyuan Cao, Zhipeng Zhang, Xing Zhu, et al. Reward Forcing: Efficient streaming video generation with rewarded distribution matching distillation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2026.

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.

Wenyang Luo, Haina Qin, Bing Li, Jiwen Lu, Xin Tao, Pengfei Wan, and Kun Gai. Improving autoregressive video modeling with history understanding. In *International Conference on Learning Representations*, 2026.

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

Alexander Quinn Nichol and Prafulla Dhariwal. Improved denoising diffusion probabilistic models. In *International Conference on Machine Learning*, pp. 8162–8171. PMLR, 2021.

William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 4195–4205, 2023.

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.

Haonan Qiu, Menghan Xia, Yong Zhang, Yingqing He, Xintao Wang, Ying Shan, and Ziwei Liu. FreeNoise: Tuning-free longer video diffusion via noise rescheduling. *arXiv preprint arXiv:2310.15169*, 2023.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67, 2020.

Weiming Ren, Harry Yang, Ge Zhang, Cong Wei, Xinrun Du, Stephen Huang, and Wenhu Chen. ConsistI2V: Enhancing visual consistency for image-to-video generation. *arXiv preprint arXiv:2402.04324*, 2024.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10684–10695, 2022.

Tim Salimans and Jonathan Ho. Progressive distillation for fast sampling of diffusion models. In *International Conference on Learning Representations*, 2022.

Sand.ai, Hansi Teng, Hongyu Jia, Lei Sun, Lingzhi Li, Maolin Li, Mingqiu Tang, Shuai Han, Tianning Zhang, W. Q. Zhang, et al. MAGI-1: Autoregressive video generation at scale. *arXiv preprint arXiv:2505.13211*, 2025.

Axel Sauer, Dominik Lorenz, Andreas Blattmann, and Robin Rombach. Adversarial diffusion distillation. *arXiv preprint arXiv:2311.17042*, 2023.

Uriel Singer, Adam Polyak, Thomas Hayes, Xi Yin, Jie An, Songyang Zhang, Qiyuan Hu, Harry Yang, Oron Ashual, Oran Gafni, et al. Make-A-Video: Text-to-video generation without text-video data. *arXiv preprint arXiv:2209.14792*, 2022.

Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. In *International Conference on Learning Representations*, 2021a.

Yang Song, Jascha Sohl-Dickstein, Diederik P. Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. In *International Conference on Learning Representations*, 2021b.

Yang Song, Prafulla Dhariwal, Mark Chen, and Ilya Sutskever. Consistency models. In *International Conference on Machine Learning*, 2023.

Zachary Teed and Jia Deng. RAFT: Recurrent all-pairs field transforms for optical flow. In *European Conference on Computer Vision*, pp. 402–419, 2020.

Thomas Unterthiner, Sjoerd van Steenkiste, Karol Kurach, Raphael Marinier, Marcin Michalski, and Sylvain Gelly. Towards accurate generative models of video: A new metric and challenges. *arXiv preprint arXiv:1812.01717*, 2019.

Team Wan, Ang Wang, Baole Ai, Bin Wen, Chaojie Mao, Chen-Wei Xie, Di Chen, Feiwu Yu, Haiming Zhao, Jianxiao Yang, et al. Wan: Open and advanced large-scale video generative models. *arXiv preprint arXiv:2503.20314*, 2025.

Qianqian Wang, Yen-Yu Chang, Ruojin Cai, Zhengqi Li, Bharath Hariharan, Aleksander Holynski, and Noah Snavely. Tracking everything everywhere all at once. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023.

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.

Zhouxia Wang, Ziyang Yuan, Xintao Wang, Yaowei Li, Tianshui Chen, Menghan Xia, Ping Luo, and Ying Shan. MotionCtrl: A unified and flexible motion controller for video generation. In *ACM SIGGRAPH*, 2024.

Thaddäus Wiedemer, Yuxuan Li, Paul Vicol, Shixiang Shane Gu, Nick Matarese, Kevin Swersky, Been Kim, Priyank Jaini, and Robert Geirhos. Video models are zero-shot learners and reasoners. *arXiv preprint arXiv:2509.20328*, 2025.

Xindi Wu, Despoina Paschalidou, Jun Gao, Antonio Torralba, Laura Leal-Taixé, Olga Russakovsky, Sanja Fidler, and Jonathan Lorraine. Motion attribution for video generation. *arXiv preprint arXiv:2601.08828*, 2026.

Jinbo Xing, Menghan Xia, Yong Zhang, Haoxin Chen, Wangbo Yu, Hao Liu, Gongye Wang, Tien-Tsin Wong, Ying Shan, and Xintao Wang. DynamiCrafter: Animating open-domain images with video diffusion priors. In *European Conference on Computer Vision*, 2024.

Jiazheng Xu, Xiao Liu, Yuchen Wu, Yuxuan Tong, Qinkai Li, Ming Ding, Jie Tang, and Yuxiao Dong. ImageReward: Learning and evaluating human preferences for text-to-image generation. *Advances in Neural Information Processing Systems*, 36:15903–15935, 2023.

Wilson Yan, Yunzhi Zhang, Pieter Abbeel, and Aravind Srinivas. VideoGPT: Video generation using VQ-VAE and transformers. *arXiv preprint arXiv:2104.10157*, 2021.

An Yang, Baosong Yang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Zhou, et al. Qwen2 technical report. *arXiv preprint arXiv:2407.10671*, 2024a.

An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, et al. Qwen2.5 technical report. *arXiv preprint arXiv:2412.15115*, 2024b.

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. In *International Conference on Learning Representations*, 2026.

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*, 2024c.

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. *arXiv preprint arXiv:2303.12346*, 2023.

Tianwei Yin, Michaël Gharbi, Taesung Park, Richard Zhang, Eli Shechtman, Fredo Durand, and Bill Freeman. Improved distribution matching distillation for fast image synthesis. *Advances in Neural Information Processing Systems*, 37:47455–47487, 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.

Tianwei Yin, Qiang Zhang, Richard Zhang, William T Freeman, Fredo Durand, Eli Shechtman, and Xun Huang. From slow bidirectional to fast autoregressive video diffusion models. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 22963–22974, 2025.

Lijun Yu, Yong Cheng, Kihyuk Sohn, José Lezama, Han Zhang, Huiwen Chang, Alexander G. Hauptmann, Ming-Hsuan Yang, Yuan Hao, Irfan Essa, and Lu Jiang. MAGVIT: Masked generative video transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10459–10469, 2023.

Lijun Yu, José Lezama, Nitesh Bharadwaj Gundavarapu, Luca Versari, Kihyuk Sohn, David Minnen, Yong Cheng, Agrim Gupta, Xiuye Gu, Alexander G. Hauptmann, Boqing Gong, Ming-Hsuan Yang, Irfan Essa, David Ross, and Lu Jiang. Language model beats diffusion – tokenizer is key to visual generation. In *International Conference on Learning Representations*, 2024.

APPENDIX

A IMPLEMENTATION DETAILS

Implementation and training. We implement DynPrompt-Driven with Wan2.1-T2V-1.3B (Wan et al., 2025) as our base model, which generates 5-second videos at 16 FPS with a resolution of 832×480. Training prompts are sampled from a filtered and LLM-extended version of VidProM (Wang & Yang, 2024). Denoising is applied chunk-wise using 3 latent frames per chunk.

Training proceeds in three stages. First, following CausVid (Yin et al., 2025), we initialize the model with causal attention masking and train it on 16k ODE solution pairs sampled from the base model. Next, we follow the Self-Forcing training setup (Huang et al., 2025) for 600 iterations with a batch size of 64. Finally, we incorporate TPR and MGGA and continue training for another 600 iterations with a batch size of 8. The short-video ablations in Table 3 start from the checkpoint after the first two stages. The configuration without either module is also fine-tuned for 600 iterations with a batch size of 8 and serves as the baseline in the module ablation.

We use the AdamW optimizer for both the generator G_θ, with a learning rate of 1.5×10^{-6}, and the fake model s_{fake}, with a learning rate of 4.0×10^{-7}. The generator is updated every 5 steps of fake score updates. We use AllTracker (Harley et al., 2025) to extract motion masks for motion-aware training.

Short-video evaluation protocol. Following Self Forcing (Huang et al., 2025), we generate 5-second videos using 946 official VBench (Huang et al., 2024a;b) prompts rewritten by Qwen/Qwen2.5-7B-Instruct (Yang et al., 2024b), each sampled with 5 different random seeds to ensure statistical reliability and reduce variance in evaluation results.

Prompt recalibration ablation protocol. We compare three prompt-conditioning configurations in Table 3b: (i) **Static Prompt**, which uses the original fixed text embedding throughout all autoregressive steps; (ii) **Fixed-interval Switching**, where we employ Qwen/Qwen2.5-7B-Instruct (Yang et al., 2024b) to decompose the original prompt into two semantically coherent sub-prompts, and switch from the first to the second at a predetermined midpoint during generation; and (iii) **Dynamic Prompt (Ours)**, which continuously recalibrates the prompt embedding conditioned on the accumulated visual KV-cache.

Noise schedule and model parameterization. Following the Wan2.1 series, we adopt the flow matching framework (Lipman et al., 2022; Liu et al., 2022), with time step shifting

$$t'(k,t) = \frac{kt/1000}{1 + (k-1)(t/1000)} \cdot 1000 \tag{10}$$

and a shift factor $k = 5$. The forward process is specified as

$$x_t = \frac{1000 - t'}{1000} x + \frac{t'}{1000}\epsilon, \quad \epsilon \sim \mathcal{N}(0, I), \quad t \in [0, 1000]. \tag{11}$$

Thus, $t = 1000$ corresponds to pure noise ($x_{1000} = \epsilon$), while $t = 0$ corresponds to clean data ($x_0 = x$). With the velocity convention $v = \epsilon - x$, the clean-data prediction is:

$$G_\theta(x_t, t, c) = x_t - \frac{t'}{1000}\, v_\theta(x_t, t', c). \tag{12}$$

This follows by rearranging $x_t = x + (t'/1000)(\epsilon - x)$. Our few-step diffusion process employs a uniform 4-step schedule before applying the above time shift:

$$[t_4,\, t_3,\, t_2,\, t_1] = [1000,\, 750,\, 500,\, 250]. \tag{13}$$

B EXTENDED VBENCH EVALUATION

We report quantitative evaluation on VBench (Huang et al., 2024a) using the extended prompts. Each dimension is first normalized according to the official VBench ranges before the weighted

aggregation below. Specifically, the Quality Score is a weighted average of the following dimensions: subject consistency, background consistency, temporal flickering, motion smoothness, aesthetic quality, imaging quality, and dynamic degree. The Semantic Score is a weighted average of the following dimensions: object class, multiple objects, human action, color, spatial relationship, scene, appearance style, temporal style, and overall consistency.

Table 4: **Quality evaluation on extended VBench.** SC: Subject Consistency, BC: Background Consistency, TF: Temporal Flickering, MS: Motion Smoothness, DD: Dynamic Degree, AQ: Aesthetic Quality, IQ: Imaging Quality, QS: Quality Score, Tot: Total Score.

Model	SC	BC	TF	MS	DD	AQ	IQ	QS	Tot
CausVid	96.33	95.84	99.44	97.98	61.11	64.52	67.96	83.93	82.88
Self Forcing	95.09	96.10	99.01	98.24	66.38	65.79	69.71	84.59	83.80
LongLive	96.98	96.92	99.35	98.79	40.83	67.03	69.18	83.68	83.22
Reward Forcing	95.43	96.59	98.97	98.32	68.05	65.66	69.38	84.84	84.13
Ours	94.23	95.61	98.76	98.64	83.88	65.09	69.33	**85.62**	**84.60**

Table 5: **Semantic evaluation on extended VBench.** OC: Object Class, MO: Multiple Objects, HA: Human Action, Co: Color, SR: Spatial Relationship, Sc: Scene, TS: Temporal Style, AS: Appearance Style, OvC: Overall Consistency, SS: Semantic Score, Tot: Total Score.

Model	OC	MO	HA	Co	SR	Sc	TS	AS	OvC	SS	Tot
CausVid	92.78	88.32	96.20	86.67	74.05	51.35	23.95	20.19	25.95	78.69	82.88
Self Forcing	93.16	87.19	96.40	86.83	81.77	56.13	24.45	20.34	26.85	80.64	83.80
LongLive	96.28	86.49	95.80	90.79	80.56	58.79	24.16	20.42	26.61	**81.37**	83.22
Reward Forcing	94.81	86.79	96.80	89.42	82.47	57.19	24.33	20.38	26.88	81.32	84.13
Ours	94.14	85.41	96.60	86.36	80.17	56.67	24.58	20.48	26.95	80.53	**84.60**

C Motion-Guided Gradient Alignment: Additional Details

Reference video library construction. We curate a high-dynamic reference video library $\mathcal{V}_{\mathrm{Q}}$ using Veo 3 (Google DeepMind, 2025; Wiedemer et al., 2025). The library covers 10 motion categories: *walking/running*, *jumping*, *spinning/rotating*, *bouncing*, *flowing liquid*, *fire/explosion*, *falling objects*, *waving/flapping*, *deforming/morphing*, and *crowd motion*. Each category contains 10 videos of 8-second clips at 24 FPS with a resolution of 1280×720, yielding $|\mathcal{V}_{\mathrm{Q}}| = 100$ reference clips in total. The prompts favor simple backgrounds to reduce background-related interference; some also specify camera motion. Ten example query prompts are listed below:

- **Stride:** A horse gallops at full speed across an open flat wilderness; smooth side-tracking camera, neutral daylight, clean background. Emphasis on the lifting and landing of hooves, the rippling of muscles, and the flowing of the mane.
- **Roll:** A car tire rolls forward along a straight asphalt road; smooth side-tracking camera, bright natural light, minimalist road background. Emphasis on the continuous contact of the rubber tread and the angular momentum of rotation.
- **Spin:** A coin is flicked on a flat tabletop and spins rapidly in place; close-up static low-angle camera, side lighting, seamless background. Emphasis on the visual motion blur of the coin face and the low-frequency wobble as it approaches a stop.
- **Bounce:** A ping-pong ball bounces on a white table; stable side-view camera, neutral lighting, seamless background. Emphasis on consistent bounce height and timing.
- **Slide:** A granite curling stone glides slowly forward along a smooth ice lane; smooth frontal low-angle tracking camera, uniform indoor lighting, clean ice background. Emphasis on the extremely smooth deceleration and a slight rotational trajectory.
- **Swing:** A metal pendulum swings rhythmically left and right in the air; static frontal camera, bright studio lighting, solid background. Emphasis on the arc trajectory under gravity and the momentary pause at the highest point.

- **Free Fall:** An apple drops vertically from a tree branch onto flat grass; stable side-view camera, bright natural light, minimalist background. Emphasis on the increasing velocity due to gravitational acceleration and the instantaneous stop upon impact.
- **Stretch:** A pair of hands vigorously knead a soft lump of dough on a flat wooden board; close-up static top-down camera, neutral lighting, clean background. Emphasis on the compression indentation, tensile elongation, and plastic deformation of the dough.
- **Explode:** A watermelon bursts apart the instant it is struck by a heavy object; close-up static camera, high-speed slow-motion capture, neutral lighting, solid background. Emphasis on the tearing of the rind, the shattering of the flesh, and the splashing of the juice.
- **Flow:** Thick orange-red lava flows downward along a gentle slope; stable top-down camera, low-key ambient light to accentuate the self-luminous glow of the lava, solid background. Emphasis on the cooling and crusting of the surface, the churning of the interior, and the viscous fluid morphology.

Gradient alignment implementation. AllTracker outputs anchor-to-target displacement fields (trajectories). We convert them into per-frame motion magnitudes:

$$\Delta \mathbf{f}_t = \mathbf{f}_t - \mathbf{f}_{t-1}, \qquad m_t(x,y) = \|\Delta \mathbf{f}_t(x,y)\|_2, \qquad m_0 \equiv 0, \tag{14}$$

where $\mathbf{f}_0$ is assigned zero motion magnitude. We then splat the anchor-frame motion magnitudes onto the target-frame coordinate system to avoid misalignment between the mask and the current frame content:

$$M_{\text{target},t}(x',y') = \frac{\sum_{(x,y)} m_t(x,y)\, w_t(x,y)\, I_t(x,y;x',y')}{\sum_{(x,y)} w_t(x,y)\, I_t(x,y;x',y') + \epsilon}, \tag{15}$$

where $I_t(x,y;x',y')$ is one when the rounded displaced coordinates $(\lfloor x + f_{t,x}(x,y) \rceil, \lfloor y + f_{t,y}(x,y) \rceil)$ equal (x',y'), and zero otherwise. The weight $w_t(x,y) = v_t(x,y) \cdot v_{t-1}(x,y)$ is the product of per-pixel visibility confidences from AllTracker ($w_t \equiv 1$ when confidence is disabled). Writing $M = M_{\text{target},t}$, the splatted map is then min–max normalized into a spatial weight map:

$$W_{\text{motion}}(x,y) = \frac{M(x,y) - \min M}{\max M - \min M + \epsilon}. \tag{16}$$

Directly projecting a gradient vector with a dense random matrix costs $O(d \cdot k)$, where d is the number of gradient coordinates and k is the projection dimension. The blockwise Fastfood-based transform (Le et al., 2013) described below costs $O(\lceil d/d_F \rceil d_F \log d_F)$ per projection, or $O(d \log d_F)$ when $d \geq d_F$; T projections incur this cost T times.

Projection matrix construction (pre-computed at initialization). For each projection $i \in \{1, \ldots, T\}$ we draw:

- $\mathbf{b}_i \in \{-1, +1\}^{d_F}$: a Rademacher random vector;
- $\boldsymbol{\pi}_i$: a uniformly random permutation of $[d_F]$;
- $\mathbf{g}_i \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_{d_F})$: a Gaussian scaling vector;
- $\mathbf{s}_i = |\mathbf{z}_i| + \epsilon\mathbf{1}$, with $\mathbf{z}_i \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_{d_F})$: a positive scaling vector,

where $d_F = 2^{\lceil \log_2 k \rceil}$ is the smallest power of two no less than k, as required by the Walsh–Hadamard transform (WHT).

Single projection transform. Given a gradient vector $\mathbf{x}$, we partition it into blocks of size d_F (zero-padding the last block if necessary) and apply to each block:

$$P(\mathbf{x}) = \frac{1}{\xi}\, \mathbf{S}\, H\, \mathbf{G}\, \boldsymbol{\Pi}\, H\, \mathbf{B}\, \mathbf{x}, \tag{17}$$

where H denotes the normalized WHT with scaling factor $d_F^{-1/2}$, $\xi = d_F^{1/2}$ is the variance normalisation constant, and $\mathbf{B}$, $\mathbf{G}$, and $\mathbf{S}$ are diagonal matrices induced by $\mathbf{b}_i$, $\mathbf{g}_i$, and $\mathbf{s}_i$, respectively. $\boldsymbol{\Pi}$ is

the permutation matrix induced by π_i. The WHT is implemented via the butterfly algorithm with complexity $O(d_F \log d_F)$.

To examine the association between gradient alignment and prompt motion semantics, we conduct an additional analysis during training. For each training sample, we record the cosine similarity between the fake-model gradient direction induced by the locally generated video and that induced by the sampled high-dynamic query video, following the MGGA formulation. Specifically, for a local video and a query video, we compute their gradient directions with respect to the fake model and save the corresponding cosine similarity score. A higher score indicates that the local sample induces a parameter-update direction more consistent with the high-dynamic query sample.

In parallel, we estimate the motion propensity of each text prompt using Qwen/Qwen2.5-7B-Instruct (Yang et al., 2024b). Given a prompt, the language model is asked to classify its semantics into one of two categories: static-oriented or dynamic-oriented. We then examine whether samples whose prompts are classified as dynamic-oriented also receive high MGGA cosine similarity scores. Concretely, we collect all saved cosine similarity scores during training and use the median score as a threshold, corresponding to the top 50% high-alignment samples. For all samples whose prompts are labeled as dynamic-oriented by Qwen, we compute the hit rate of their cosine similarity scores falling into this top-50% group. The resulting hit rate is 83.7%, indicating that most prompts with stronger motion semantics indeed correspond to local videos whose gradient directions are more aligned with high-dynamic query videos. This result supports an association between prompt motion semantics and gradient alignment. It does not directly measure the accuracy of distinguishing dynamic from static generated videos.

D Further Related Works

Video diffusion models. Diffusion models have become the dominant paradigm for image and video generation, with foundational formulations including DDPM, DDIM, score-based modeling, and classifier-free guidance (Ho et al., 2020; Song et al., 2021a;b; Nichol & Dhariwal, 2021; Dhariwal & Nichol, 2021; Ho & Salimans, 2022). Latent diffusion models improve efficiency via compressed latent spaces (Rombach et al., 2022). Building on these, video diffusion models extend denoising to the temporal dimension for text-conditioned synthesis (Ho et al., 2022b;a; Singer et al., 2022; Blattmann et al., 2023; Bar-Tal et al., 2024), with large-scale systems such as Movie Gen, Wan, CogVideoX, HunyuanVideo, and SkyReels demonstrating strong scaling for high-resolution and long-form generation (Polyak et al., 2024; Wan et al., 2025; Lin et al., 2024; Yang et al., 2024c; Kong et al., 2024; Chen et al., 2025). Video controllability and motion enhancement have also been explored in image-to-video and trajectory-conditioned settings (Xing et al., 2024; Wang et al., 2024; Ren et al., 2024; Guo et al., 2024). However, such bidirectional context is fundamentally at odds with the causal structure required for real-time streaming generation.

E User Study Details

The user study recruited 20 volunteers (12 male, 8 female; ages 22–35; all with normal or corrected-to-normal vision). For each compared method, 20 videos were randomly sampled from both our method and the baseline, and each volunteer was asked to rate all compared methods. Videos generated by all methods were presented in randomized order without method labels. Volunteers rated each video on two dimensions: *Motion Richness* (0–10) and *Visual Quality* (0–10). The study was conducted online using a standardized interface with fixed display resolution and video playback speed.

The 20 prompts span diverse motion categories. Motion Richness evaluates whether motion is sufficient and natural, while Visual Quality evaluates perceptual quality and temporal consistency. The endpoints of the rating scale are 0 (poor dynamics or visually unsatisfactory content) and 10 (rich dynamics with high visual fidelity). Figure 4 visualizes the reported ratings.

F Long-video Evaluation Protocol

VBench-Long evaluation. Table 2 compares Rolling Forcing, Rolling Forcing with our method, Reward Forcing, and Reward Forcing with our method on 60-second videos. We use the official

VBench-Long evaluator (Huang et al., 2024b) in `long_custom_input` mode with `dev_flag` enabled and the default metric settings. Each video is divided into non-overlapping two-second clips. We evaluate subject consistency, background consistency, motion smoothness, dynamic degree, aesthetic quality, and imaging quality, using the evaluator's aggregation for each dimension. The six aggregate scores are multiplied by 100 for presentation in Table 2, with higher values indicating better performance.

Quality drift. Following Reward Forcing (Lu et al., 2026), we quantify temporal variation in imaging quality using the 30 two-second clips of each 60-second video. For each clip, we use its raw MUSIQ imaging-quality score, averaged over its frames. We compute the sample standard deviation of the 30 chronologically ordered clip scores within each video, with a degrees-of-freedom correction of one. The reported Drift is the arithmetic mean of these per-video standard deviations over the evaluated videos. Drift is expressed in the same units as the raw MUSIQ scores, without additional scaling. Lower Drift indicates smaller fluctuations in imaging quality across the video.

F.1 Long-video Component Ablation

Table 6 compares the baseline, TPR alone, MGGA alone, and their combination on 60-second videos, using Rolling Forcing as the backbone. We use the same seven evaluation metrics and aggregation protocol as in Table 2.

TPR and MGGA individually increase dynamic degree from 27.00 to 36.44 and 34.21, respectively; combining them raises it to 44.35 and achieves the lowest Drift of 2.15. TPR alone increases Drift from 2.22 to 2.43, indicating that its motion gain does not by itself ensure more stable imaging quality over time.

Table 6: **Component ablation on 60-second videos.** All configurations use Rolling Forcing as the backbone. Higher is better for the six VBench-Long dimensions; lower is better for Drift.

Configuration	VBench-Long evaluation scores ↑						Drift ↓
	Subject Consistency	Background Consistency	Motion Smoothness	Dynamic Degree	Aesthetic Quality	Imaging Quality	
Base	98.23	96.66	98.94	27.00	61.12	71.34	2.22
Base + TPR	98.15	96.45	98.77	36.44	61.54	70.84	2.43
Base + MGGA	97.87	96.33	98.54	34.21	60.98	71.61	2.18
Base + TPR + MGGA	97.98	96.27	98.76	44.35	62.47	70.98	2.15

G Long-video Qualitative Comparisons

Figure 5 presents 60-second examples for Rolling Forcing and Reward Forcing, each paired with its adaptation using our method.

H Limitations and Future Work

While DynPrompt-Driven demonstrates substantial improvements in motion dynamics for autoregressive video generation, several limitations remain that point to promising directions for future work.

Dependence on high-dynamic reference library. Motion-Guided Gradient Alignment relies on a curated reference set $\mathcal{V}_{\mathrm{Q}}$ of 100 high-dynamic videos synthesized by Veo 3, covering 10 motion categories. Although we designed these categories to span diverse motion modalities, the coverage remains finite, and the quality of gradient alignment is inherently bounded by the representativeness of this library. Videos with rare or compositionally complex motion patterns—such as multi-agent interactions or fine-grained articulated motion—may not be adequately represented, potentially limiting the generalization of Motion-Guided Gradient Alignment to such scenarios. Future work

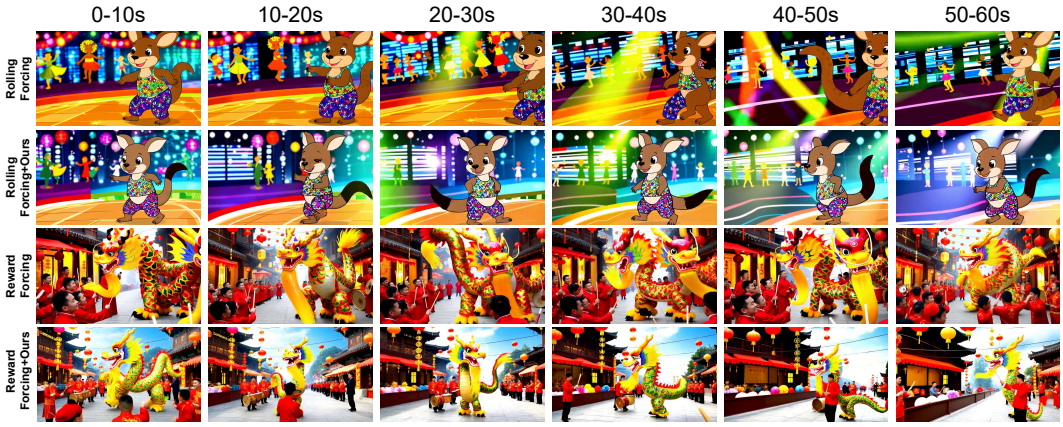

Figure 5: **Qualitative comparison of long-video generation.** The top two rows compare Rolling Forcing and Rolling Forcing with our method; the bottom two rows compare Reward Forcing and Reward Forcing with our method.

could explore automatically expanding the reference library via online mining of high-dynamic samples from large-scale video datasets, or replacing the static reference set with a dynamically updated memory bank that adapts to the evolving distribution of generated videos during training.

Evaluation on longer video generation. Our long-video evaluation measures visual quality, consistency, dynamics, and drift for adaptations to Rolling Forcing and Reward Forcing (Table 2); the diagnostics in Figure 1 further examine motion and prompt conditioning over 60-second rollouts. These metrics do not directly measure adherence to the actions specified in the prompt. Systematic evaluation beyond one minute and assessment of motion persistence together with long-range visual quality remain directions for further study.

Distinction from offline motion-aware data selection. Wu et al. (2026) use motion-aware gradient attribution to select an offline subset of training videos, then fine-tune the generator on that subset. Our Motion-Guided Gradient Alignment uses related gradient-alignment information for online reweighting: each iteration samples a high-motion query video and weights generated samples' DMD losses by their gradient alignment with the query. In our DMD stage, reference videos serve to compute these weights and are not used as direct supervised targets. The distinction is therefore in how the attribution signal is used: selecting an offline fine-tuning dataset versus weighting generated samples during distribution matching.

Generalization to other autoregressive architectures. Our primary implementation uses Wan2.1-T2V-1.3B with a self-forcing-style distillation pipeline. The adaptations to Rolling Forcing and Reward Forcing in Table 2 provide additional evidence within autoregressive video diffusion. Broader validation across backbones with different attention mechanisms, tokenization strategies, and distillation objectives remains necessary.