Enhancing Train-Free Infinite-Frame Generation for Consistent Long Videos

Xiaokun Feng [1 2 3] **Jiashu Zhu** [3] **Meiqi Wu** [3] **Chubin Chen** [3]
Fangyuan Mao [3] **Haiyang Guo** [3] **Jiahong Wu** [3 ‡] **Xiangxiang Chu** [3] **Kaiqi Huang** [1 2 †]

Abstract

Without incurring significant computational overhead, train-free long video generation aims to enable foundation video generation models to produce longer videos. Frame-level autoregressive frameworks, *e.g.*, FIFO-diffusion, offer the advantage of generating infinitely long videos with constant memory consumption. However, the mismatch between training and inference, coupled with the challenge of maintaining long-term consistency, limits the effective utilization of foundation models. To mitigate these concerns, we propose **MIGA**, a novel infinite-frame long video generation method. Firstly, we propose an effective two-stage alignment mechanism that mitigates the training-inference gap by reducing the excessive noise span fed to the model. We then introduce an innovative dual consistency enhancement mechanism, where the self-reflection approach corrects early high-noise frames and the long-range frame guidance approach leverages later low-noise frames with broad coverage to steer generation, jointly improving temporal consistency. Extensive experiments on VBench and NarrLV demonstrate the state-of-the-art performance of MIGA. Our project page is available at https://xiaokunfeng.github.io/miga_homepage/.

1. Introduction

Recent advances in video generation (Wan et al., 2025; Yang et al., 2024) have demonstrated impressive capabilities in synthesizing short video clips. However, many real-world applications, such as film production, game development, and world simulation, require coherent long video generation (Cho et al., 2024; Mao et al., 2026). Building long video generation models from scratch typically requires substantial computational and data resources, owing to the inherent complexity of the video modality (Waseem & Shahzad, 2024; Hu et al., 2023). Given the remarkable performance of off-the-shelf foundation video generation models on short videos (Chen et al., 2024b; Wan et al., 2025), a more efficient and practical approach is to extend their generation length in a training-free manner (Qiu et al., 2023).

To achieve train-free long video generation, one straightforward strategy is to increase the number of latents fed into foundation models and design specific mechanisms that transfer their short-term generation capabilities to long video scenarios. For example, FreeNoise (Qiu et al., 2023) ingeniously reorganizes the initial noise and models temporal dependencies via window-based fusion. Following this paradigm, FreeLong (Lu et al., 2024) and FreePCA (Tan et al., 2025) integrate the global and local information from the perspectives of frequency and principal component analysis, respectively. Although these methods have shown promising results, their memory requirements increase proportionally with the number of generated frames, which significantly restricts the achievable video length (*e.g.*, generating minute-long videos).

To enable infinite frame generation, recent studies such as Diffusion-Forcing (Chen et al., 2024a) and AR-Diffusion (Sun et al., 2025) attempt to assign different noise levels to different latent features, thereby empowering diffusion models to iteratively generate in an autoregressive fashion. Notably, FIFO-Diffusion (Kim et al., 2024) maintains a noise queue where noise levels increase sequentially along the frame dimension, and employs a first-in-first-out denoising process for frame-level autoregressive generation. More importantly, this approach requires only fixed memory consumption, enabling FIFO-Diffusion to support infinite-frame generation.

Despite these merits, train-free frame-level autoregressive models such as FIFO-Diffusion still leave considerable room for further improvement. On one hand, a substantial gap

Work done during the internship at AMAP, Alibaba Group. † Corresponding author ‡ Project lead [1]School of Artificial Intelligence, University of Chinese Academy of Sciences, Beijing, China [2]The Key Laboratory of Cognition and Decision Intelligence for Complex Systems, Institute of Automation, Chinese Academy of Sciences, Beijing, China [3]AMAP, Alibaba Group, Beijing, China. Correspondence to: Kaiqi Huang <kaiqi.huang@nlpr.ia.ac.cn>.

Proceedings of the 43rd International Conference on Machine Learning, Seoul, South Korea. PMLR 306, 2026. Copyright 2026 by the author(s).

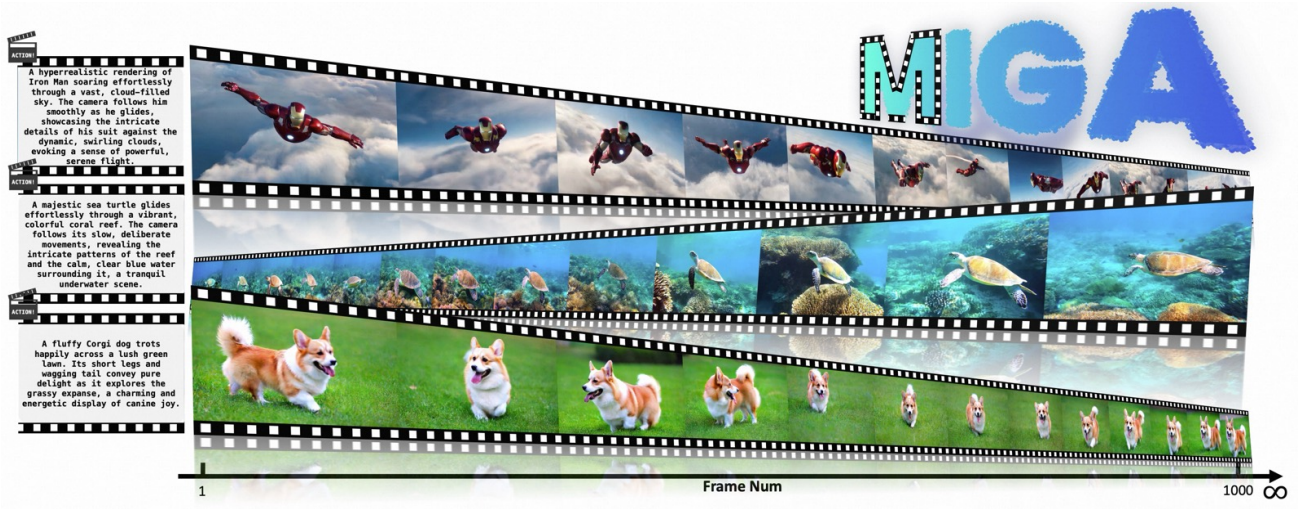


Figure 1. **MIGA enables temporally consistent, infinite-frame (∞) video generation in a training-free manner.** We present three long videos (1000+ frames) generated by MIGA, while the foundation model used by MIGA, Wan2.1-1.3B (Wan et al., 2025), supports only 81 frames by default.

exists between training and inference in long video generation (Kim et al., 2024). In particular, during training, the model is exposed to input latents with a single noise level, whereas during inference, it must handle multiple noise levels corresponding to the number of frames. These discrepancies prevent the foundation generation models from fully realizing their potential, which in turn leads to issues such as content drift and visual artifacts (Cai et al., 2025). On the other hand, long-term consistency is a central objective for long video generation (Henschel et al., 2025; Yang et al., 2025a), yet existing methods pay insufficient attention to this goal. For example, FIFO-Diffusion only facilitates feature interaction between neighboring chunks through lookahead denoising, but lacks explicit modeling of long-range frame dependencies, resulting in suboptimal long video quality.

To address these limitations, we propose **MIGA**, a novel train-free method for infinite-frame video generation. Firstly, we propose an intuitive and effective **two-stage training-inference alignment mechanism** to mitigate the inherent training-inference gap in existing train-free autoregressive frameworks. As this gap primarily arises from the excessive noise span of latents fed to the model during inference, we alleviate it through two dedicated optimization stages. The first stage maintains a zigzag-structured latent queue to proactively narrow the noise span of input latents. In the second stage, once all latents are denoised to the same noise level, a unified denoising process is conducted, achieving a noise span that matches that of the training phase.

Furthermore, leveraging the properties of the maintained long latents queue, we present an innovative **dual consistency enhancement mechanism** to promote long-term consistency. For early high-noise latents, we design a self-reflection approach that efficiently evaluates and promptly corrects them, thereby ensuring consistency in the subsequently generated video. Unlike existing methods that rely on external evaluators and redundant computations (Yang et al., 2025a; He et al., 2025), our approach achieves this solely through self-similarity analysis among early latents. For the later low-noise latents, we introduce a long-range frame guidance approach that incorporates them into each denoising iteration, facilitating feature interactions between distant frames. Benefiting from these improvements, MIGA achieves significant gains of 4.7% and 2.0% in subject and background consistency on VBench (Huang et al., 2024), respectively, compared to FIFO-Diffusion with a similar framework. Moreover, evaluations on NarrLV (Feng et al., 2025b) demonstrate that MIGA exhibits exceptional capability in generating rich narrative content.

In summary, our contributions are as follows:

- To inherit the merits of train-free frame-level autoregressive frameworks while alleviating their limitations in training-inference gap and long-term consistency modeling, we propose a novel infinite-frame generation method, MIGA.
- We design an effective two-stage training-inference alignment mechanism that proactively mitigates the training-inference gap by optimizing the noise span. Furthermore, we introduce an innovative dual consistency enhancement mechanism that promotes long-term consistency through self-reflection and long-range frame guidance.
- Comprehensive experiments on the mainstream

VBench and NarrLV benchmarks demonstrate that MIGA achieves new state-of-the-art performance.

Conflict of Interest Disclosure. The authors X.F., J.Z., M.W., C.C., F.M., H.G., J.W., and X.C. are employed by AMAP, Alibaba Group. One of the open-source foundation models used in this work, Wan2.1 (Wan et al., 2025), was developed by the Tongyi Lab of Alibaba Group, a separate team independent of AMAP. The authors had no privileged access to Wan2.1 beyond its public open-source release. All experiments follow the standard public benchmarks (VBench and NarrLV) and evaluation protocols to ensure fair comparison.

2. Related Works

Text-to-Video Generation. Recently, the field of video generation has witnessed remarkable advancements. Early approaches primarily adopted frameworks that combine 2D spatial and 1D temporal modeling, such as VideoCrafter (Chen et al., 2023; 2024b), and Stable Video Diffusion (Blattmann et al., 2023). These have progressively transitioned into more advanced 3D full-attention architectures, as illustrated by Video Diffusion Models (Ho et al., 2022) and CogVideoX (Yang et al., 2024). Recently developed foundation models, including HunyuanVideo (Kong et al., 2024) and Wan (Wan et al., 2025) have further contributed to the improvement in video quality (Huang et al., 2024; Ling et al., 2025) . Despite offering more accessible tools for video generation, current video diffusion models are generally constrained to training on short, fixed-length videos (Lu et al., 2024; Lu & Yang, 2025; Tan et al., 2025). Given the crucial role of long videos in practical scenarios (Cho et al., 2024), achieving consistent generation of long videos has emerged as a prominent research topic.

Long Video Generation. In pursuit of long video generation, several studies (Yan et al., 2025; Guo et al., 2025b; Teng et al., 2025; Chen et al., 2025c; Xiao et al., 2025; Deng et al., 2024; Huang et al., 2025) introduce specialized architectures and perform large-scale training on curated datasets. However, their heavy reliance on computational and data resources limits broad adoption within the community (Lu et al., 2024). To address this, recent work explores train-free strategies to efficiently extend the output duration of foundation video generators in a resource-friendly manner. For example, Gen-L-Video (Wang et al., 2023) extends video length by merging overlapping subsequences with a sliding-window method. FreeNoise (Qiu et al., 2023), FreeLong (Lu et al., 2024), and FreePCA (Tan et al., 2025) integrate local and global features by leveraging discovered patterns in initialization noise, frequency distributions, and principal component structures. RIFLEx (Zhao et al., 2025) refines temporal position encodings to reduce periodic repetition. Unlike these finite-extension methods, FIFO-Diffusion (Kim et al., 2024) equips diffusion models with frame-level autoregressive generation via a noise space design (Chen et al., 2024a; Liu et al., 2025b), supporting infinite frames with fixed memory. Building on this, we propose MIGA to retain autoregressive advantages while addressing the training-inference gap and consistency limitations.

3. Methods

3.1. Preliminaries: Train-Free Frame-Level Autoregressive Generation.

Mainstream diffusion-based video generation models typically comprise a conditional encoder (*e.g.*, a text encoder), a variational autoencoder (VAE), and a noise prediction network $\varepsilon_\theta(\cdot)$. The VAE enables bidirectional mapping between pixel-level video data and compact latents, $\mathbf{z}_0 = [\mathbf{z}_0^1; ...; \mathbf{z}_0^f] \in \mathbb{R}^{f\times l\times d}$, where f, l, and d represent the frame count, tokens per frame, and token dimension, respectively. For clarity, we regard the latent feature of each frame (*e.g.*, $\mathbf{z}_0^i \in \mathbb{R}^{l\times d}, i \in [1, f]$) as a basic unit throughout this paper. For example, the number of latents in $\mathbf{z}_0$ is f. Given a trained $\varepsilon_\theta(\cdot)$, the fully denoised latents $\mathbf{z}_0$ can be recovered from Gaussian noise $\mathbf{z}_T \sim \mathcal{N}(0, I)$. Following a time step schedule $0 = \tau_0 < \tau_1 < \cdots < \tau_T = T$, $\mathbf{z}_0$ is generated by progressively refining $\mathbf{z}_{\tau_T} = [\mathbf{z}_{\tau_T}^1; \ldots; \mathbf{z}_{\tau_T}^f]$ over T steps with a sampler $\phi(\cdot)$ (*e.g.*, DDPM (Ho et al., 2020)). Each denoising step is formulated as:

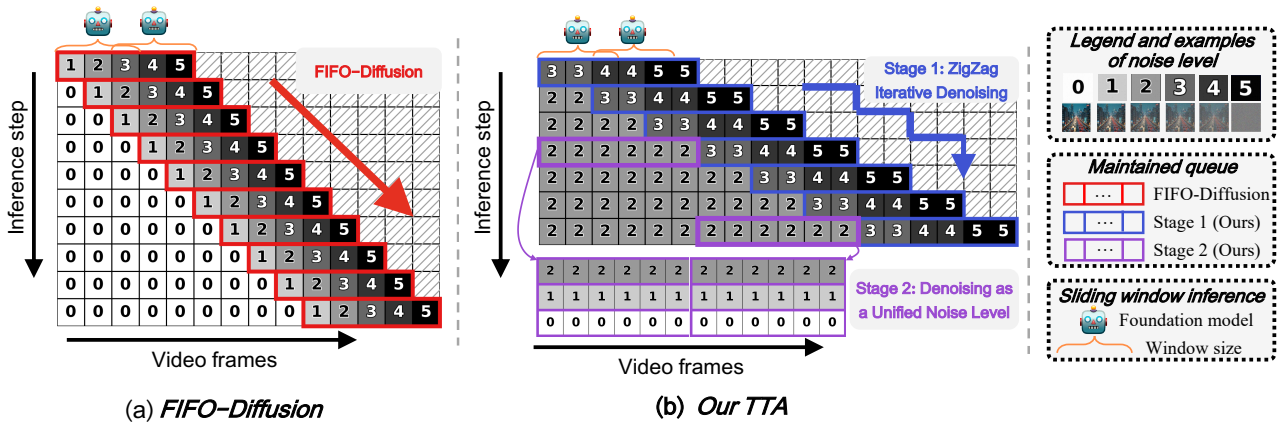
$$[\mathbf{z}_{\tau_{t-1}}^1; \ldots; \mathbf{z}_{\tau_{t-1}}^f] = \phi([\mathbf{z}_{\tau_t}^1; \ldots; \mathbf{z}_{\tau_t}^f], [\tau_t; \ldots; \tau_t]; \varepsilon_\theta), \tag{1}$$

where $\mathbf{z}_{\tau_t}^i$ denotes the latent of the i-th frame at time step τ_t. For convenience, conditional inputs (*e.g.*, text prompts) are omitted in the above formulation.

To enable a foundation model that can only generate f_0 frames to produce long videos consisting of N frames ($N \gg f_0$), frame-level autoregressive generation centers around maintaining a latents queue $\mathcal{Q} = \{\mathbf{z}_{\tau_1}^1; \ldots; \mathbf{z}_{\tau_T}^T\}$, which contains T latents (*i.e.*, its length L equals the total number of denoising steps T) with progressively increasing noise level, as illustrated in Fig. 2 (a). After applying one inference step to all latents in $\mathcal{Q}$:

$$\{\mathbf{z}_{\tau_0}^1; \ldots; \mathbf{z}_{\tau_{T-1}}^T\} = \Phi(\{\mathbf{z}_{\tau_1}^1; \ldots; \mathbf{z}_{\tau_T}^T\}, \{\tau_1; \ldots; \tau_T\}; \varepsilon_\theta), \tag{2}$$

the first latent in the queue, $\mathbf{z}_{\tau_0}^1$, becomes a fully denoised, clean latent. By dequeuing $\mathbf{z}_{\tau_0}^1$ from $\mathcal{Q}$ and appending a new Gaussian latent $\mathbf{z}_{\tau_T}^T$ to the end, the process can be repeated to realize frame-level autoregressive generation. In this way, FIFO-Diffusion realizes a diagonal denoising paradigm. Notably, since the queue length T is typically greater than the number of frames f_0 that the model can process, a single inference step of the sampler $\Phi(\cdot)$ over $\mathcal{Q}$



(a) *FIFO-Diffusion* (b) *Our TTA*

Figure 2. **Inference framework comparison between FIFO-Diffusion and our Two-Stage Training-Inference Alignment (TTA) mechanism. (a)** FIFO-Diffusion achieves frame-level autoregressive generation by maintaining a queue of latents with progressively increasing noise levels, resulting in an excessive noise span among the local latents fed to the model. **(b)** Our TTA effectively reduces the noise span: Stage 1 performs zigzag denoising by slowing down the rate of noise changes, and Stage 2 applies unified denoising once the output latents reach the same noise level.

involves multiple executions of the standard sampler $\phi(\cdot)$. For instance, FIFO-Diffusion employs a sliding window approach with a window size of f_0 and a stride of $\lfloor f_0/2 \rfloor$. Prior to performing the above autoregressive generation, $\mathcal{Q}$ must be properly initialized. For details of initialization and the autoregressive generation procedure, please refer to App. A.1.1.

3.2. Two-Stage Training-Inference Alignment (TTA).

The effectiveness of the aforementioned train-free frame-level autoregressive generation relies on the assumption that the foundation model can perform noise prediction on f_0 latents with varying noise levels. However, the model is trained to denoise f_0 latents at unified noise levels. This significant gap between training and inference hinders the foundation model's full generative potential. Although FIFO-Diffusion (Kim et al., 2024) has considered this issue and theoretically proved that the error introduced by train-free autoregressive generation is bounded by the span of noise levels, its final approach still requires the model to handle latents with a noise span of f_0. Given the impact of the noise span in input latents, a natural question emerges: ***can we further reduce the noise span of latents fed to the model during inference, so as to better align the input condition with that of training?*** Motivated by this, we decompose the generation process into two stages, aiming to maximally align training and inference by intuitively and effectively reducing the noise span.

Stage 1: Zigzag Iterative Denoising. Autoregressive generation inherently requires maintaining a noise queue that inevitably covers a range of noise levels (Chen et al., 2024a; Sun et al., 2025). To reduce the noise span of latents processed by the model, an intuitive adjustment is to slow down the rate at which noise levels change within the queue. Specifically, as shown in Fig. 2 (b), we initialize and maintain a noise queue $\mathcal{Q}_{s_1}$ as follows:

$$\mathcal{Q}_{s_1} = \{\underbrace{\mathbf{z}^1_{\tau_e}, \cdots, \mathbf{z}^{L_{\text{zig}}}_{\tau_e}}_{L_{\text{zig}}}, \underbrace{\mathbf{z}^{L_{\text{zig}}+1}_{\tau_{e+1}}, \cdots, \mathbf{z}^{2L_{\text{zig}}}_{\tau_{e+1}}}_{L_{\text{zig}}}, \cdots, \underbrace{\mathbf{z}^{L-L_{\text{zig}}+1}_{\tau_T}, \cdots, \mathbf{z}^{L}_{\tau_T}}_{L_{\text{zig}}}\}. \tag{3}$$

Unlike existing methods that change the noise level with every single latent frame, our queue alters it every L_{zig} latents. This zigzag structure provides the model with a smoother noise span across inputs, contributing to mitigating the training–inference gap. At each iteration, we dequeue the first L_{zig} latents $\mathbf{z}^i_{\tau_e}$ (where $i \in [1, L_{\text{zig}}]$) from the front of the queue, and append L_{zig} new Gaussian latents $\mathbf{z}^T_{\tau_T}$ to its end. It is important to note that the time step τ_e of the first L_{zig} latents in the queue is greater than τ_0, which means that Stage 1 only partially completes the denoising process. The subsequent denoising steps are carried out in Stage 2.

Stage 2: Denoising at a Unified Noise Level. After n iterations in Stage 1, we obtain nL_{zig} latents, all at the same time step τ_{e-1}. These latents form the queue $\mathcal{Q}_{s_2}$ to be processed in the second stage:

$$\mathcal{Q}_{s_2} = \{\mathbf{z}^1_{\tau_{e-1}}, \mathbf{z}^2_{\tau_{e-1}}, \ldots, \mathbf{z}^{nL_{\text{zig}}}_{\tau_{e-1}}\}. \tag{4}$$

Since all latent frames share the same noise level, the model processes latents with identical intensity at each denoising operation. This setup aligns well with the conditions seen during training. We also apply the sliding-window denoising over $\mathcal{Q}_{s_2}$, sequentially processing its frames. As the foundation model handles a fixed latent length per pass, memory

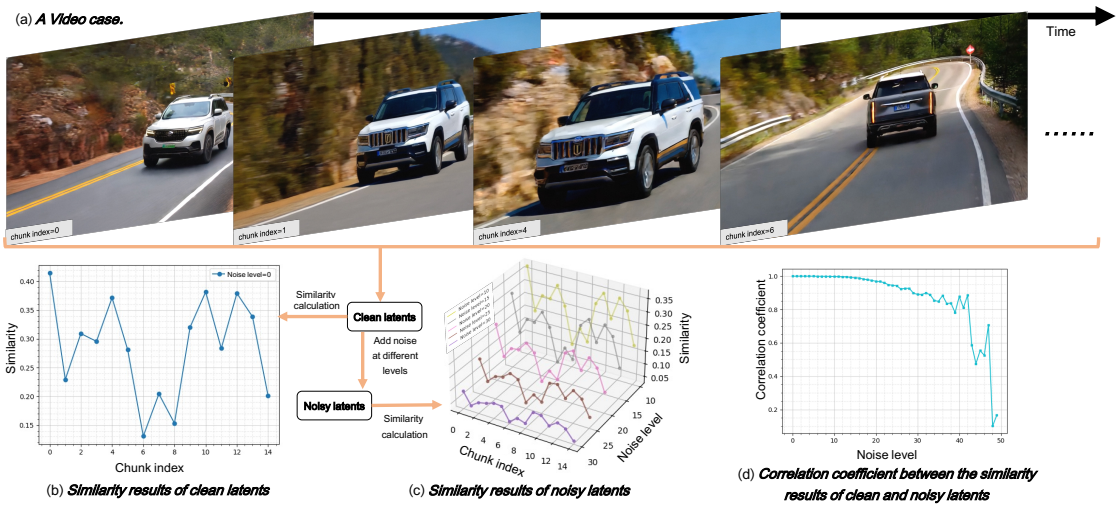


Figure 3. **Modeling insight behind our self-reflection mechanism. (a)** A video case containing the consistency anomaly. **(b-d)** Similarity computation between clean and noisy latents, along with the corresponding correlation coefficient analysis results.

usage does not grow with longer videos. After $(e-1)$ iterative denoising steps, we obtain nL_{zig} fully denoised frames (*i.e.*, $N = nL_{\text{zig}}$ frames in the generated video). Details of the TTA procedure are provided in App. A.1.2.

3.3. Dual Consistency Enhancement (DCE).

Although the TTA mechanism effectively mitigates the gap between training and inference, it still lacks dedicated modeling designs for the crucial goal of long-term generation tasks, *i.e.*, maintaining long-term consistency. To address this issue, we propose an innovative dual consistency enhancement mechanism based on the characteristics of our maintained latent queue. Specifically, the **self-reflection approach** focuses on latents at the tail of the queue, efficiently evaluating and correcting newly added latents. Besides, the **long-range frame guidance approach** targets latents at the head of the queue, incorporating long-range, low-noise latents into each local denoising process. The roles of these two methods within the queue are shown in the framework diagram in Fig. A1 of App. A.1.3.

Self-Reflection. Recent advances in LLMs (Guo et al., 2025a; Jaech et al., 2024; Bai et al., 2023) have explored test-time scaling (TTS) (Zhang et al., 2025; Chen et al., 2026), which improves response quality by allocating extra computation during inference. Inspired by this, TTS techniques have been adapted to video generation (Ma et al., 2025; Liu et al., 2025a; He et al., 2025; Wu et al., 2025; Chen et al., 2025a), using multiple candidate latents with evaluation and selection strategies to enhance video quality. Unlike prior work focusing on fixed-length videos, our self-reflection approach integrates TTS with the characteristics of frame-level autoregressive generation for long videos. Given that temporal consistency is crucial in long video generation, long-term consistency is typically set as the primary objective when extending the search time. The most relevant work, ScalingNoise (Yang et al., 2025a), uses a consistency reward for long video generation. Differences between our approach and ScalingNoise, as well as other TTS methods, are detailed below.

Our self-reflection approach interprets TTS as comprising two processes: first, adaptively evaluating the locations where anomalies (*e.g.*, abrupt drops in consistency) occur; second, performing an expanded search at these points for correction (please refer to Fig. A1 (b) for the flowchart). Unlike previous methods that either conduct search at every step (Yang et al., 2025a) or at predefined scheduler steps (He et al., 2025), our approach aims to efficiently and flexibly determine when to trigger expanded search. To achieve this, a straightforward and accurate consistency metric is required. Existing methods often rely on external models for quantitative assessment. For example, ScalingNoise uses DINO (Caron et al., 2021) for consistency evaluation, introducing redundancy into the pipeline. Moreover, since these models require clean pixel inputs, consistency assessment during intermediate denoising steps necessitates additional denoising and VAE decoding procedures (Yang et al., 2025a), resulting in high computational overhead. To overcome these limitations, we propose a more efficient consistency evaluation strategy inspired by the following observation.

Firstly, the latent space produced by the VAE, after large-scale pre-training, exhibits strong interpretability (Kingma & Welling, 2013). Specifically, the distance between latents reflects the degree of difference between the corresponding video frames. Therefore, we can leverage the cosine similarity between different latents as a consistency metric,

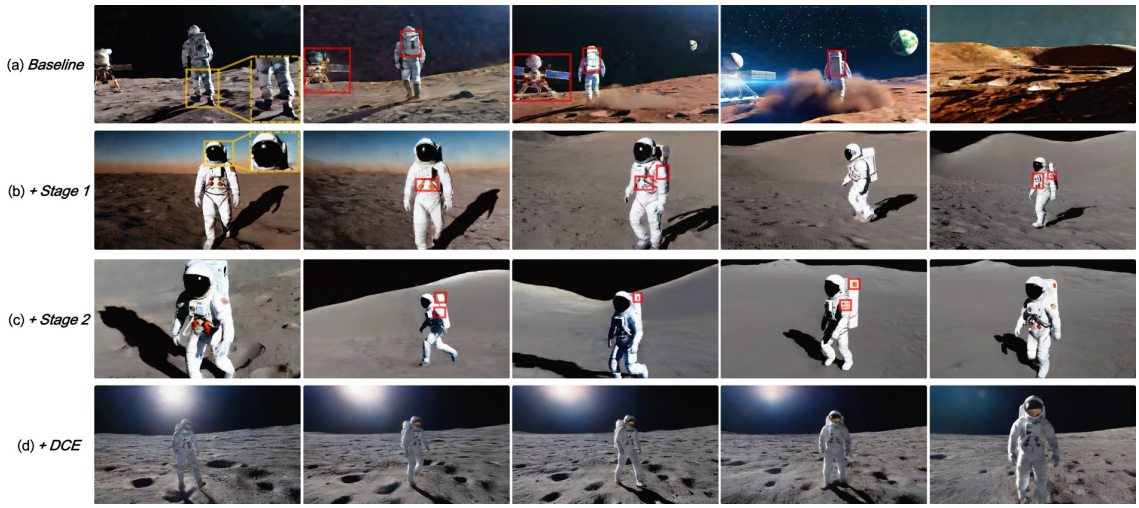


Figure 4. **Illustration of ablation study results.** (a-d) Starting from the *baseline*, our *Stage 1*, *Stage 2*, and *DCE* mechanism are sequentially added. Yellow bboxes in the first frame indicate regions with prominent noise. Red bboxes denote regions in the current frame where the subject exhibits noticeable inconsistency compared to previous frames. Better viewed in color with zoom-in.

thereby avoiding the need for additional external evaluation models. Formally, let $\mathbf{q}_{\text{eval}} \in \mathbb{R}^{f_{\text{eval}} \times l \times c}$ denote the f_{eval} consecutive latents to be evaluated, and $\mathbf{q}_{\text{ref}} \in \mathbb{R}^{f_{\text{ref}} \times l \times c}$ denote those of the preceding f_{ref} adjacent latents. The consistency score C_{score} is computed as follows:

$$\mathbf{q}'_{\text{eval}} = \text{norm}_1\left(\text{mean}_2(\mathbf{q}_{\text{eval}})\right), \tag{5}$$

$$\mathbf{q}'_{\text{ref}} = \text{norm}_1\left(\text{mean}_2(\mathbf{q}_{\text{ref}})\right), \tag{6}$$

$$C_{\text{score}} = \text{mean}_1\left(\text{mean}_2\left(\mathbf{q}'_{\text{eval}}\mathbf{q}'^{\,\text{T}}_{\text{ref}}\right)\right), \tag{7}$$

where $\text{norm}_i(\cdot)$ and $\text{mean}_i(\cdot)$ denote normalization and mean operations along the i-th dimension ($i \geq 0$), respectively, and T denotes matrix transpose. Fig. 3(a,b) show sequential consistency evaluation over video segments ($f_{\text{eval}} = 4$, $f_{\text{ref}} = 8$) on a clean video with a consistency anomaly. It is evident that the proposed metric can effectively identify the position of the consistency disruption.

However, considering the influence of the initial noisy latents (Qiu et al., 2023) on the final generated content—such as the overall layout of the video being largely determined in the early denoising stages—we aim to assess consistency at the early high-noise latents rather than at the clean latents, in order to enable timely adjustments. A straightforward solution is to fully denoise these high-noise latents and then compute C_{score}. Undoubtedly, such frequent denoising operations would incur substantial computational overhead. Fortunately, we observe that the early high-noise latents and the final clean latents exhibit a strong correlation in terms of C_{score}. As shown in Fig. 3 (c), higher noise levels reduce the absolute magnitude of the C_{score} curve, yet the fluctuation patterns remain similar across different noise levels. Fig. 3 (d) presents the correlation coefficients between the C_{score} curves under various noise levels and those of clean latents, indicating strong correlation even at higher noise intensities (*e.g.*, 40, with the maximum noise level being 50).

Leveraging this insight, we can timely evaluate consistency at early stages. Specifically, for the latent queue $\mathcal{Q}$ covering all noise levels, we define a judgment index f_{judg} at its tail (*i.e.*, the early high-noise latents). At each iteration over $\mathcal{Q}$, latents in $[f_{\text{judg}} - f_{\text{ref}},\ f_{\text{judg}} - 1]$ serve as references to evaluate those in $[f_{\text{judg}},\ f_{\text{judg}} + f_{\text{eval}} - 1]$. When the decrease in C_{score} between adjacent chunks exceeds the threshold δ_{adju}, an expanded search is triggered for correction.

Upon detecting a consistency anomaly, all latents after the position f_{judg}, $q^{\text{init}}_{\text{samp}} = \{z^i\}_{i=1}^{L-f_{\text{judg}}+1}$, are considered for correction (*i.e.*, the required format for our search samples). Benefiting from our early determination of the judgment node, the number of latents included in the search sample is relatively small. Taking into account the diversity and effectiveness of the search samples, we design a progressively guided search strategy. Assuming the number of search samples is n_{samp}, we first randomly sample n_{samp} Gaussian noise latents as starting points for each sample: $q^k_{\text{samp}} = \{z^1\},\ k \in [1, n_{\text{samp}}]$. Subsequently, we use the preceding f_{guid} latents before f_{judg}, which have passed our evaluation, as the guiding information, denoted as $q_{\text{guid}} = \{z^i\}_{i=1}^{f_{\text{guid}}}$. For each sample q^k_{samp}, we concatenate it with q_{guid} and perform iterative denoising. After each iteration, only the latents in q^k_{samp} are updated, and a new noise frame is appended to q^k_{samp}. After $L - f_{\text{judg}}$ iterations, q^k_{samp} contains $(L - f_{\text{judg}} + 1)$ latents, maintaining the same data format as q^{init}_{samp}.

After obtaining k candidate samples, the consistency score

C^k_{score} is computed for each candidate based on its first f_{eval} latents. If the maximum score exceeds $C^{\text{init}}_{\text{score}}$, $Q^{\text{init}}_{\text{samp}}$ is replaced by the corresponding Q^k_{samp}, completing the correction. Details of the sampling and correction process are provided in Alg. 6 in App. A.1.3.

Long-Range Frame Guidance. When applying the foundation model for sliding inference on the maintained queue $\mathcal{Q} = \{z^i\}_{i=1}^L$, the model only accesses f_0 nearby latents at each denoising step. To enable interactions among distant latents (Feng et al., 2024) and thereby enhance the consistency of generated videos, we propose a simple yet effective long-range frame guidance method. Specifically, when the model processes latents within a local window, we explicitly sample m_{guid} latents sparsely from earlier positions in $\mathcal{Q}$. Since these latents are relatively clean, we utilize them to guide the denoising of the current latents (see Fig. A1 (a) for an illustration). Formally, when the model slides to position $l \in [1, m_{\text{guid}}]$, the input is defined as $q_{\text{input}} = [z^l, \dots, z^{l+f_0-1}]$. When $l \in (m_{\text{guid}},\, L - m_{\text{guid}}]$, we have

$$q_{\text{input}} = [z^1, \dots, z^{m_{\text{guid}}},\, z^l, \dots, z^{l+f_0-m_{\text{guid}}-1}]. \quad (8)$$

for head latents in the queue (*i.e.*, $l \leq m_{\text{guid}}$), long-range guidance is not applied due to the insufficient number of preceding latents. Note that these latents are also obtained by iterative denoising propagated from the tail of the sequence, such that they have also been guided by their preceding latents during the process. For the selection of m_{guid} guidance latents, we uniformly sample m_{guid} latents from the $\min(m_{\text{guid}} L_{\text{zig}}, l-1)$ prior latents before position l.

4. Experiments

4.1. Implementation Details.

Model Implementation. To ensure fair comparison with prior training-free long video generation methods, we apply MIGA to the widely-used foundation model, VideoCrafter2 (Chen et al., 2024b). Besides, we incorporate MIGA into the latest model, Wan2.1-1.3B (Wan et al., 2025). By default, these models generate 16 and 21 latents, respectively. For infinite-frame generation, the configuration of VideoCrafter2-based MIGA is as follows: $T = 64$, $L_{\text{zig}} = 4$, $\tau_e = 10$, $\delta_{\text{adju}} = 0.01$, and $m_{\text{guid}} = 6$. And Wan2.1-based MIGA adopts $T = 54$, $L_{\text{zig}} = 7$, $\tau_e = 10$, $\delta_{\text{adju}} = 0.01$, and $m_{\text{guid}} = 4$. Moreover, thanks to the flexibility of the frame-level autoregressive framework, we can achieve multi-text control by providing different text conditions to the latents at various temporal positions. For more implementation details, see App. A.3 and App. A.4.

Evaluation Benchmarks. Our evaluations are conducted on the VBench (Huang et al., 2024) and NarrLV (Feng et al., 2025b) benchmarks. Following the evaluation protocols of existing long video generators, we primarily use the video quality dimensions assessed by the VBench-Long toolkit. The commonly used metrics include subject consistency (S.C.), background consistency (B.C.), motion smoothness (M.S.), temporal flicker (T.F.), and their mean, Overall Score (O.S.). NarrLV is a recent benchmark for narrative expressiveness in long video models. We use evaluation prompts with Temporal Narrative Atom (TNA) counts of 2, 3, and 4, and report results on three dimensions, *i.e.*, scene attributes (s_{att}), target attributes (t_{att}), and target actions (t_{act}).

Table 1. Quantitative results of MIGA and baselines on VBench. The best results are highlighted in **bold**.

Method	Infinite	S.C.	B.C.	M.S.	T.F.	O.S.
VideoCrafter2-Based						
FreePCA	✗	93.57	95.24	93.73	91.27	93.45
FreeLong	✗	95.72	96.42	98.38	97.28	96.95
FIFO-Diffusion	✓	92.92	95.01	97.19	94.94	95.02
ScalingNoise	✓	94.29	95.52	97.86	96.12	95.95
MIGA (ours)	✓	**97.66**	**96.99**	**98.60**	**98.03**	**97.82**
Wan2.1-Based						
FIFO-Diffusion	✓	92.67	93.37	98.03	97.09	95.29
MIGA (ours)	✓	**96.46**	**95.50**	**98.85**	**98.14**	**97.24**

Compared Baselines. For methods that extend video length by increasing input latents, we compare against FreePCA (Tan et al., 2025) and FreeLong (Lu et al., 2024). For approaches supporting infinite-length video generation, we select FIFO-Diffusion (Kim et al., 2024) and ScalingNoise (Yang et al., 2025a) as representative baselines. Recently, training-based methods such as Self-Forcing (Huang et al., 2025) have enabled frame-level autoregressive long video generation. As these methods are beyond the scope of this work, we discuss and compare them in App B.4. Results show that MIGA achieves comparable performance to these training-based approaches.

4.2. Comparison with Baselines.

VBench. As shown in Tab. 1, MIGA achieves state-of-the-art performance across all metrics, for both foundation models. For the VideoCrafter2-based models, we standardized the generation length to 128 frames to ensure fair comparison. Compared to the strong baseline FreeLong, MIGA demonstrates further improvement in both subject and background consistency, which validates the effectiveness of our approach in enhancing video consistency. For the Wan2.1-based models, we evaluate the generation of 161-frame videos. Compared against FIFO-Diffusion, which also adopts the autoregressive framework, MIGA achieves comprehensive improvements in all metrics, highlighting the efficacy of our optimizations to the framework. It is worth noting that the consistency scores of Wan2.1-based

Table 2. Quantitative results of MIGA and baselines under varying TNA settings on NarrLV.

Method	Infinite	TNA=2			TNA=3			TNA=4		
		s_{att}	t_{att}	t_{act}	s_{att}	t_{att}	t_{act}	s_{att}	t_{att}	t_{act}
VideoCrafter2-Based										
FreePCA	✗	56.96	58.72	56.41	53.61	53.93	52.57	50.46	57.28	53.27
FreeLong	✗	59.43	59.57	55.95	56.57	59.82	56.57	54.13	60.53	54.13
ScalingNoise	✓	59.28	55.47	58.09	53.27	58.14	54.05	52.37	58.41	53.59
FIFO-Diffusion	✓	67.02	63.55	58.29	61.15	60.64	58.42	66.09	66.01	54.66
MIGA (ours)	✓	**69.78**	**63.94**	**59.01**	**63.53**	**61.05**	**59.52**	**68.87**	**68.77**	**55.78**
Wan2.1-Based										
FIFO-Diffusion	✓	67.77	64.25	65.40	55.42	59.02	58.91	57.43	56.10	53.89
MIGA (ours)	✓	**79.32**	**67.87**	**67.94**	**69.48**	**66.33**	**63.86**	**75.05**	**72.31**	**62.90**

MIGA are slightly lower than those of VideoCrafter2-based MIGA. We conjecture that this is because the latter primarily generates animation-style videos, where maintaining long-term consistency is relatively easier than in the realistic style videos produced by the former. A qualitative example and analysis are provided in App. B.2. Subsequent evaluation results on the NarrLV demonstrate that Wan2.1-based MIGA achieves notable advantages in narrative expressiveness.

NarrLV. To generate videos that correspond to the rich narrative content in NarrLV, we adopt a sequence of changing prompts guidance strategy similar to FIFO-Diffusion. Specifically, latents at different temporal positions are conditioned on different prompts. As shown in Tab. 2, FIFO-Diffusion outperforms existing VideoCrafter2-based methods in narrative expressiveness. Furthermore, our MIGA achieves additional improvements, which can be attributed to its design that enables more stable video content generation and thereby supports richer semantic expression.

Qualitative Results. Fig. 1 presents three 1k-frame videos generated by the Wan2.1-based MIGA, which closely follow the text prompts and maintain strong subject and background consistency. For additional visualizations, please refer to Fig. A6 and Fig. A7 in App. B.6.

Due to space limits, additional **user studies** and **computational efficiency analyses** are in App. B.5 and App. B.3.

4.3. Ablation Study.

To explore the effectiveness of our mechanism design, we conduct comprehensive ablation studies on the VideoCrafter2-based MIGA using VBench. Implementation details for each setting are provided in App. B.1.

Study on Core Mechanisms Designs. The core contribution of our work is the introduction of the novel Two-Stage Training-Inference Alignment (TTA) and Dual Consistency Enhancement (DCE) mechanisms. As shown in Tab. 3, FIFO-Diffusion serves as the baseline without our proposed mechanisms. Individually, TTA and DCE improve the overall score by 2.03% and 1.73%, respectively, demonstrating their effectiveness. Combined, they provide complementary gains and further enhance performance.

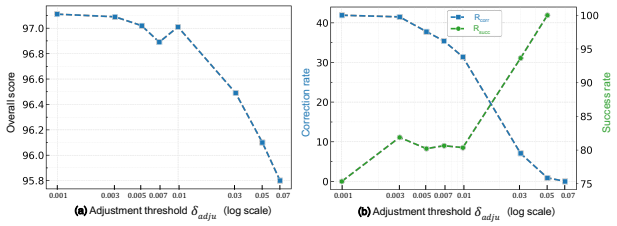


Figure 5. Ablation study on the adjustment threshold δ_{adju}. **(a-b)** Effects of δ_{adju} on O.S., R_{corr}, and R_{succ}.

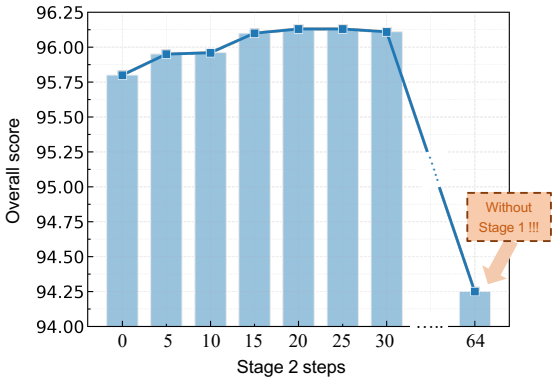


Figure 6. Ablation study on the steps in stage 2.

Study on TTA. Our TTA mechanism comprises two stages: stage 1 employs zigzag iterative denoising, and stage 2 applies denoising at a unified noise level. As shown in Tab. 6, sequentially introducing these two stages leads to notable performance improvements. As shown in Fig. 4 (a–c), stage 1 significantly reduces drastic anomalies present in the baseline, while stage 2 plays a key role in suppressing video noise. Furthermore, Tab. 4 shows the impact of the zigzag width L_{zig} in stage 1 on model performance. As L_{zig} in-

Table 3. Ablation results of our core mechanisms (*i.e.*, TTA, DCE).

TTA	DCE	S.C.	B.C.	M.S.	T.F.	O.S.
		92.92	95.01	97.19	94.94	95.02
✓		96.74	96.75	97.57	97.12	97.05
	✓	96.10	96.47	97.88	96.56	96.75
✓	✓	97.66	96.99	98.60	98.03	97.82

Table 4. Ablation results of L_{zig}.

L_{zig}	S.C.	B.C.	M.S.	T.F.	O.S.
1	94.23	94.52	97.98	96.47	95.80
2	94.24	95.93	98.55	97.90	96.66
4	95.37	95.96	98.65	98.02	97.00
6	95.14	96.04	98.60	97.97	96.94
8	95.54	95.96	98.56	97.90	96.99

Table 5. Ablation results of m_{guid}.

m_{guid}	S.C.	B.C.	M.S.	T.F.	O.S.
0	94.23	94.52	97.98	96.47	95.80
2	94.66	94.72	98.64	98.05	96.52
4	94.59	94.58	98.64	98.10	96.48
6	95.45	95.69	98.45	97.89	96.87
8	95.32	95.12	98.60	98.05	96.77

Table 6. Ablation results of TTA design.

Setting	S.C.	B.C.	M.S.	T.F.	O.S.
Baseline	92.92	95.01	97.19	94.94	95.02
+Stage 1	95.98	96.54	97.57	97.03	96.78
+Stage 2	96.74	96.75	97.57	97.12	97.05

creases, *i.e.*, the foundation model processes frames within a smaller noise span, the Overall Score (O.S.) first improves and then saturates. Accordingly, we set $L_{\text{zig}} = 4$. Additionally, Fig. 6 illustrates the effect of varying the number of second-stage steps, *i.e.*, $(e-1)$. As the step count increases, performance gains gradually stabilize. However, if only the second stage is performed, *i.e.*, inference is conducted directly on N noise latents, model performance drops sharply. This approach is no longer autoregressive, as it requires simultaneous processing of independently initialized latents. In contrast, the omitted stage 1 leverages autoregressive generation to establish implicit information transfer among the initialized latents. Further discussion is in App. B.1.

Study on DCE. Our DCE mechanism consists of self-reflection and long-range frame guidance. Fig. 5 shows the impact of different adjustment thresholds δ_{adju}, where a smaller δ_{adju} leads to more frequent searches. Fig. 5 (b) presents two metrics that reflect the search effectiveness. We denote the total inference steps, steps evaluated for correction, and steps actually corrected as n_{all}, n_{eval}, and n_{corr}, respectively. The correction rate and success rate are defined as $R_{\text{corr}} = n_{\text{corr}}/n_{\text{all}}$ and $R_{\text{succ}} = n_{\text{corr}}/n_{\text{eval}}$. As the threshold decreases, the model tends to perform broader searches, increasing the corrected steps (*i.e.*, R_{corr}) and consequently improving the O.S. metric, demonstrating test-time scaling capability. Since the number of steps with consistency anomalies within n_{all} is limited, performance gains eventually converge. Further reducing the threshold increases n_{eval} but not n_{corr}, leading to a lower R_{succ}. Balancing performance and computational cost, we set the default δ_{adju} to 0.01. Tab. 5 presents the effect of varying the number of guidance frames, with optimal performance achieved at $m_{\text{guid}} = 6$. App. B.1 provides an analysis of computational efficiency.

5. Conclusion

Train-free frame-level autoregressive generation frameworks, represented by FIFO-Diffusion, demonstrate the capability to generate infinitely long videos with constant memory cost. To build on these merits while overcoming limitations in the training-inference gap and long-term consistency, we introduced MIGA, a novel infinite-frame generation method. An effective two-stage training-inference alignment mechanism is developed to proactively mitigate the training-inference gap by optimizing the noise span. Additionally, an innovative dual consistency enhancement mechanism is proposed to improve long-term consistency through self-reflection and long-range frame guidance. Experiments show that MIGA achieves state-of-the-art results on VBench and NarrLV compared to existing train-free methods, demonstrating its ability to generate long videos with strong temporal consistency and rich narratives.

Acknowledgements

This work is supported in part by the National Science and Technology Major Project (Grant No.2022ZD0116403).

Impact Statement

This paper presents work whose goal is to advance the field of Machine Learning. There are many potential societal consequences of our work, none which we feel must be specifically highlighted here.

References

Bai, J., Bai, S., Chu, Y., Cui, Z., Dang, K., Deng, X., Fan, Y., Ge, W., Han, Y., Huang, F., et al. Qwen technical report. *arXiv preprint arXiv:2309.16609*, 2023.

Bai, Z., Wang, P., Xiao, T., He, T., Han, Z., Zhang, Z., and Shou, M. Z. Hallucination of multimodal large language models: A survey. *arXiv preprint arXiv:2404.18930*, 2024.

Bansal, H., Lin, Z., Xie, T., Zong, Z., Yarom, M., Bitton, Y., Jiang, C., Sun, Y., Chang, K.-W., and Grover, A.

Videophy: Evaluating physical commonsense for video generation. *arXiv preprint arXiv:2406.03520*, 2024.

Blattmann, A., Dockhorn, T., Kulal, S., Mendelevitch, D., Kilian, M., Lorenz, D., Levi, Y., English, Z., Voleti, V., Letts, A., et al. Stable video diffusion: Scaling latent video diffusion models to large datasets. *arXiv preprint arXiv:2311.15127*, 2023.

Cai, S., Yang, C., Zhang, L., Guo, Y., Xiao, J., Yang, Z., Xu, Y., Yang, Z., Yuille, A., Guibas, L., et al. Mixture of contexts for long video generation. *arXiv preprint arXiv:2508.21058*, 2025.

Caron, M., Touvron, H., Misra, I., Jégou, H., Mairal, J., Bojanowski, P., and Joulin, A. Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 9650–9660, 2021.

Chen, B., Martí Monsó, D., Du, Y., Simchowitz, M., Tedrake, R., and Sitzmann, V. Diffusion forcing: Next-token prediction meets full-sequence diffusion. *Advances in Neural Information Processing Systems*, 37:24081–24125, 2024a.

Chen, C., Hu, S., Zhu, J., Wu, M., Chen, J., Li, Y., Huang, N., Fang, C., Wu, J., Chu, X., et al. Taming preference mode collapse via directional decoupling alignment in diffusion reinforcement learning. *arXiv preprint arXiv:2512.24146*, 2025a.

Chen, C., Zhu, J., Feng, X., et al. S^2-guidance: Stochastic self guidance for training-free enhancement of diffusion models. *arXiv preprint arXiv:2508.12880*, 2025b.

Chen, G., Lin, D., Yang, J., Lin, C., Zhu, J., Fan, M., Zhang, H., Chen, S., Chen, Z., Ma, C., et al. Skyreels-v2: Infinite-length film generative model. *arXiv preprint arXiv:2504.13074*, 2025c.

Chen, H., Xia, M., He, Y., Zhang, Y., Cun, X., Yang, S., Xing, J., Liu, Y., Chen, Q., Wang, X., et al. Videocrafter1: Open diffusion models for high-quality video generation. *arXiv preprint arXiv:2310.19512*, 2023.

Chen, H., Zhang, Y., Cun, X., Xia, M., Wang, X., Weng, C., and Shan, Y. Videocrafter2: Overcoming data limitations for high-quality video diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7310–7320, 2024b.

Chen, H., Lou, X., Feng, X., Huang, K., and Wang, X. Unveiling chain of step reasoning for vision-language models with fine-grained rewards. *Advances in Neural Information Processing Systems*, 38:114703–114727, 2026.

Cho, J., Puspitasari, F. D., Zheng, S., Zheng, J., Lee, L.-H., Kim, T.-H., Hong, C. S., and Zhang, C. Sora as an agi world model? a complete survey on text-to-video generation. *arXiv preprint arXiv:2403.05131*, 2024.

Chu, Z., Zhang, L., Sun, Y., Xue, S., Wang, Z., Qin, Z., and Ren, K. Sora detector: A unified hallucination detection for large text-to-video models. *arXiv preprint arXiv:2405.04180*, 2024.

Deng, H., Pan, T., Diao, H., Luo, Z., Cui, Y., Lu, H., Shan, S., Qi, Y., and Wang, X. Autoregressive video generation without vector quantization. *arXiv preprint arXiv:2412.14169*, 2024.

Esser, P., Kulal, S., Blattmann, A., Entezari, R., Müller, J., Saini, H., Levi, Y., Lorenz, D., Sauer, A., Boesel, F., et al. Scaling rectified flow transformers for high-resolution image synthesis. In *Forty-first international conference on machine learning*, 2024.

Feng, X., Li, X., Hu, S., Zhang, D., Wu, M., Zhang, J., Chen, X., and Huang, K. Memvlt: Vision-language tracking with adaptive memory-based prompts. *Advances in Neural Information Processing Systems*, 37:14903–14933, 2024.

Feng, X., Hu, S., Li, X., Zhang, D., Wu, M., Zhang, J., Chen, X., and Huang, K. Atctrack: Aligning target-context cues with dynamic target states for robust vision-language tracking. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 19850–19861, 2025a.

Feng, X., Yu, H., Wu, M., Hu, S., Chen, J., Zhu, C., Wu, J., Chu, X., and Huang, K. Narrlv: Towards a comprehensive narrative-centric evaluation for long video generation models. *arXiv preprint arXiv:2507.11245*, 2025b.

Feng, X., Zhang, D., Hu, S., Li, X., Wu, M., Zhang, J., Chen, X., and Huang, K. Cstrack: Enhancing rgb-x tracking via compact spatiotemporal features. *arXiv preprint arXiv:2505.19434*, 2025c.

Guo, D., Yang, D., Zhang, H., Song, J., Zhang, R., Xu, R., Zhu, Q., Ma, S., Wang, P., Bi, X., et al. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025a.

Guo, Y., Yang, C., Yang, Z., Ma, Z., Lin, Z., Yang, Z., Lin, D., and Jiang, L. Long context tuning for video generation. *arXiv preprint arXiv:2503.10589*, 2025b.

He, H., Liang, J., Wang, X., Wan, P., Zhang, D., Gai, K., and Pan, L. Scaling image and video generation via test-time evolutionary search. *arXiv preprint arXiv:2505.17618*, 2025.

Henschel, R., Khachatryan, L., Poghosyan, H., Hayrapetyan, D., Tadevosyan, V., Wang, Z., Navasardyan, S., and Shi, H. Streamingt2v: Consistent, dynamic, and extendable long video generation from text. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 2568–2577, 2025.

Ho, J., Jain, A., and Abbeel, P. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.

Ho, J., Salimans, T., Gritsenko, A., Chan, W., Norouzi, M., and Fleet, D. J. Video diffusion models. *Advances in neural information processing systems*, 35:8633–8646, 2022.

Hu, S., Zhang, D., Feng, X., Li, X., Zhao, X., Huang, K., et al. A multi-modal global instance tracking benchmark (mgit): Better locating target in complex spatio-temporal and causal relationship. *Advances in neural information processing systems*, 36:25007–25030, 2023.

Huang, X., Li, Z., He, G., Zhou, M., and Shechtman, E. Self forcing: Bridging the train-test gap in autoregressive video diffusion. *arXiv preprint arXiv:2506.08009*, 2025.

Huang, Z., He, Y., Yu, J., Zhang, F., Si, C., Jiang, Y., Zhang, Y., Wu, T., Jin, Q., Chanpaisit, N., et al. Vbench: Comprehensive benchmark suite for video generative models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 21807–21818, 2024.

Jaech, A., Kalai, A., Lerer, A., Richardson, A., El-Kishky, A., Low, A., Helyar, A., Madry, A., Beutel, A., Carney, A., et al. Openai o1 system card. *arXiv preprint arXiv:2412.16720*, 2024.

Kang, B., Yue, Y., Lu, R., Lin, Z., Zhao, Y., Wang, K., Huang, G., and Feng, J. How far is video generation from world model: A physical law perspective. *arXiv preprint arXiv:2411.02385*, 2024.

Kim, J., Kang, J., Choi, J., and Han, B. Fifo-diffusion: Generating infinite videos from text without training. *Advances in Neural Information Processing Systems*, 37: 89834–89868, 2024.

Kingma, D. P. and Welling, M. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

Kong, W., Tian, Q., Zhang, Z., Min, R., Dai, Z., Zhou, J., Xiong, J., Li, X., Wu, B., Zhang, J., et al. Hunyuanvideo: A systematic framework for large video generative models. *arXiv preprint arXiv:2412.03603*, 2024.

Lin, M., Wang, X., Wang, Y., Wang, S., Dai, F., Ding, P., Wang, C., Zuo, Z., Sang, N., Huang, S., et al. Exploring the evolution of physics cognition in video generation: A survey. *arXiv preprint arXiv:2503.21765*, 2025.

Ling, X., Zhu, C., Wu, M., Li, H., Feng, X., Yang, C., Hao, A., Zhu, J., Wu, J., and Chu, X. Vmbench: A benchmark for perception-aligned video motion generation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 13087–13098, 2025.

Lipman, Y., Chen, R. T., Ben-Hamu, H., Nickel, M., and Le, M. Flow matching for generative modeling. *arXiv preprint arXiv:2210.02747*, 2022.

Liu, F., Wang, H., Cai, Y., Zhang, K., Zhan, X., and Duan, Y. Video-t1: Test-time scaling for video generation. *arXiv preprint arXiv:2503.18942*, 2025a.

Liu, Y., Ren, Y., Artola, A., Hu, Y., Cun, X., Zhao, X., Zhao, A., Chan, R. H., Zhang, S., Liu, R., et al. Pusa v1. 0: Surpassing wan-i2v with $500 training cost by vectorized timestep adaptation. *arXiv preprint arXiv:2507.16116*, 2025b.

Lu, Y. and Yang, Y. Freelong++: Training-free long video generation via multi-band spectralfusion. *arXiv preprint arXiv:2507.00162*, 2025.

Lu, Y., Liang, Y., Zhu, L., and Yang, Y. Freelong: Training-free long video generation with spectralblend temporal attention. *Advances in Neural Information Processing Systems*, 37:131434–131455, 2024.

Lu, Y., Zeng, Y., Li, H., Ouyang, H., Wang, Q., Cheng, K. L., Zhu, J., Cao, H., Zhang, Z., Zhu, X., et al. Reward forcing: Efficient streaming video generation with rewarded distribution matching distillation. *arXiv preprint arXiv:2512.04678*, 2025.

Ma, N., Tong, S., Jia, H., Hu, H., Su, Y.-C., Zhang, M., Yang, X., Li, Y., Jaakkola, T., Jia, X., et al. Inference-time scaling for diffusion models beyond scaling denoising steps. *arXiv preprint arXiv:2501.09732*, 2025.

Mao, F., Hao, A., Chen, J., Liu, D., Feng, X., Zhu, J., Wu, M., Chen, C., Wu, J., and Chu, X. Omni-effects: Unified and spatially-controllable visual effects generation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 7927–7935, 2026.

Qiu, H., Xia, M., Zhang, Y., He, Y., Wang, X., Shan, Y., and Liu, Z. Freenoise: Tuning-free longer video diffusion via noise rescheduling. *arXiv preprint arXiv:2310.15169*, 2023.

Song, J., Meng, C., and Ermon, S. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020.

Sun, M., Wang, W., Li, G., Liu, J., Sun, J., Feng, W., Lao, S., Zhou, S., He, Q., and Liu, J. Ar-diffusion: Asynchronous video generation with auto-regressive diffusion. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 7364–7373, 2025.

Tan, J., Yu, H., Huang, J., Xiao, J., and Zhao, F. Freepca: Integrating consistency information across long-short frames in training-free long video generation via principal component analysis. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 27979–27988, 2025.

Teng, H., Jia, H., Sun, L., Li, L., Li, M., Tang, M., Han, S., Zhang, T., Zhang, W., Luo, W., et al. Magi-1: Autoregressive video generation at scale. *arXiv preprint arXiv:2505.13211*, 2025.

Wan, T., Wang, A., Ai, B., Wen, B., Mao, C., Xie, C.-W., Chen, D., Yu, F., Zhao, H., Yang, J., et al. Wan: Open and advanced large-scale video generative models. *arXiv preprint arXiv:2503.20314*, 2025.

Wang, F.-Y., Chen, W., Song, G., Ye, H.-J., Liu, Y., and Li, H. Gen-l-video: Multi-text to long video generation via temporal co-denoising. *arXiv preprint arXiv:2305.18264*, 2023.

Waseem, F. and Shahzad, M. Video is worth a thousand images: Exploring the latest trends in long video generation. *arXiv preprint arXiv:2412.18688*, 2024.

Wu, M., Zhu, J., Feng, X., Chen, C., Zhu, C., Song, B., Mao, F., Wu, J., Chu, X., and Huang, K. Imagery-search: Adaptive test-time search for video generation beyond semantic dependency constraints. *arXiv preprint arXiv:2510.14847*, 2025.

Xiao, J., Yang, C., Zhang, L., Cai, S., Zhao, Y., Guo, Y., Wetzstein, G., Agrawala, M., Yuille, A., and Jiang, L. Captain cinema: Towards short movie generation. *arXiv preprint arXiv:2507.18634*, 2025.

Yan, X., Cai, Y., Wang, Q., Zhou, Y., Huang, W., and Yang, H. Long video diffusion generation with segmented cross-attention and content-rich video data curation. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 3184–3194, 2025.

Yang, H., Tang, F., Hu, M., Yin, Q., Li, Y., Liu, Y., Peng, Z., Gao, P., He, J., Ge, Z., et al. Scalingnoise: Scaling inference-time search for generating infinite videos. *arXiv preprint arXiv:2503.16400*, 2025a.

Yang, S., Huang, W., Chu, R., Xiao, Y., Zhao, Y., Wang, X., Li, M., Xie, E., Chen, Y., Lu, Y., et al. Longlive: Real-time interactive long video generation. *arXiv preprint arXiv:2509.22622*, 2025b.

Yang, Z., Teng, J., Zheng, W., Ding, M., Huang, S., Xu, J., Yang, Y., Hong, W., Zhang, X., Feng, G., et al. Cogvideox: Text-to-video diffusion models with an expert transformer. *arXiv preprint arXiv:2408.06072*, 2024.

Yesiltepe, H., Meral, T. H. S., Akan, A. K., Oktay, K., and Yanardag, P. Infinity-rope: Action-controllable infinite video generation emerges from autoregressive self-rollout. *arXiv preprint arXiv:2511.20649*, 2025.

Yin, T., Gharbi, M., Park, T., Zhang, R., Shechtman, E., Durand, F., and Freeman, B. Improved distribution matching distillation for fast image synthesis. *Advances in neural information processing systems*, 37:47455–47487, 2024a.

Yin, T., Gharbi, M., Zhang, R., Shechtman, E., Durand, F., Freeman, W. T., and Park, T. One-step diffusion with distribution matching distillation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 6613–6623, 2024b.

Yin, T., Zhang, Q., Zhang, R., Freeman, W. T., Durand, F., Shechtman, E., and Huang, X. From slow bidirectional to fast autoregressive video diffusion models. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 22963–22974, 2025.

Zhang, Q., Lyu, F., Sun, Z., Wang, L., Zhang, W., Hua, W., Wu, H., Guo, Z., Wang, Y., Muennighoff, N., et al. A survey on test-time scaling in large language models: What, how, where, and how well? *arXiv preprint arXiv:2503.24235*, 2025.

Zhao, M., He, G., Chen, Y., Zhu, H., Li, C., and Zhu, J. Riflex: A free lunch for length extrapolation in video diffusion transformers. *arXiv preprint arXiv:2502.15894*, 2025.

Zhao, W., Bai, L., Rao, Y., Zhou, J., and Lu, J. Unipc: A unified predictor-corrector framework for fast sampling of diffusion models. *Advances in Neural Information Processing Systems*, 36:49842–49869, 2023.

Appendix

A. Further Details on Our Method

A.1. Pseudocode Implementation

In Sec. 3, we first take FIFO-Diffusion as an example to illustrate the concrete implementation process of the train-free frame-level autoregressive generation framework (see Sec. 3.1). Next, we analyze the significant gap between training and inference present in the existing framework, thereby motivating our **Two-Stage Training-Inference Alignment (TTA) Mechanism** (see Sec. 3.2). Finally, in light of the critical importance of consistency preservation in long video generation, our proposed **Dual Consistency Enhancement (DCE) Mechanism** is described in detail (see Sec. 3.3). In this section, we provide the detailed pseudocode implementations of each method to more clearly illustrate their procedures.

A.1.1. Preliminaries: Train-Free Frame-Level Autoregressive Generation.

As described in Sec. 3.1, existing frame-level autoregressive generation methods rely on maintaining a noise queue $\mathcal{Q} = \{\mathbf{z}^1_{\tau_1}; \dots; \mathbf{z}^T_{\tau_T}\}$ with progressively increasing noise levels. Here, we take FIFO-Diffusion (Kim et al., 2024) as an example to illustrate the computation process.

The queue is first initialized, typically using clean initial latents generated by the foundation model. Since the queue length equals the denoising step count (T), which exceeds the initial latent count f_0, FIFO-Diffusion (see Alg. 1) uses the initial latents to initialize the last f_0 queue latents; the sampler (Ho et al., 2020; Lipman et al., 2022) then adds noise from τ_i to τ_j (where $j > i$)according to the scheduler-specified noise strengths:

$$\mathbf{z}_{\tau_j} \leftarrow \Phi.\text{add_noise}(\mathbf{z}_{\tau_i}, \tau_i, \tau_j). \tag{A1}$$

Next, the remaining $(T - f)$ latents are initialized by injecting varying levels of noise into the first latent.

After initializing the noise queue, iterative inference can be performed as specified in Eq. 1. As shown in Alg. 2, with each inference step, the noise level of all latents in the queue decreases by one, making the first latent clean; this latent is then dequeued from the queue and saved. To ensure continuous generation, a Gaussian noise latent is enqueued at the end of the queue at each step.

It is important to note that in Eq. 1, one inference step by the sampler over the entire queue requires multiple noise predictions from the foundation model $\epsilon_\theta(\cdot)$, since $\epsilon_\theta(\cdot)$ can only process f_0 latents at a time while the queue contains T latents ($T > f$). Specifically, FIFO-Diffusion adopts a sliding window approach: as shown in Alg. 3, the foundation model iteratively predicts noise for the queue using a window of size f_0 with a stride of l_{stride}.

Algorithm 1 Initial latents construction in FIFO-Diffusion

Input: Denoising step count T, initial latents count f_0, denoising network $\epsilon_\theta(\cdot)$, and sampler $\Phi(\cdot)$
Input: initial clean latents $q_{\text{clean}} = \{\mathbf{z}^1_{\tau_0}; \dots; \mathbf{z}^{f_0}_{\tau_0}\}$, time steps $t_{\text{clean}} = \{\tau_0, \dots, \tau_0\}$
Output: $\mathcal{Q}_{\text{init}} = \{\mathbf{z}^1_{\tau_1}; \dots; \mathbf{z}^T_{\tau_T}\}$, $t_{\text{init}} = \{\tau_1, \dots, \tau_T\}$
$\mathcal{Q}_{\text{init}} \leftarrow \{\}$, $t_{\text{init}} \leftarrow \{\}$
 # Initialize the early latents using the first frame
for $i = 1$ **to** $T - f_0$ **do**
  $\mathbf{z}^i_{\tau_i} \leftarrow \Phi.\text{add_noise}(q_{\text{clean}}[0], t_{\text{clean}}[0], \tau_i)$
  $\mathcal{Q}_{\text{init}}.\text{enqueue}(\mathbf{z}^i_{\tau_i})$
  $t_{\text{init}}.\text{enqueue}(\tau_i)$
end for
 # Manually add noise to the initial latents
for $i = 1$ **to** f_0 **do**
  $\mathbf{z}^i_{\tau_{T-f_0+i}} \leftarrow \Phi.\text{add_noise}(q_{\text{clean}}[i], t_{\text{clean}}[i], \tau_{T-f_0+i})$
  $\mathcal{Q}_{\text{init}}.\text{enqueue}(\mathbf{z}^i_{\tau_{T-f_0+i}})$
  $t_{\text{init}}.\text{enqueue}(\tau_{T-f_0+i})$
end for
Return $\mathcal{Q}_{\text{init}}$, t_{init}

Algorithm 2 Frame-level autoregressive generation in FIFO-Diffusion

Input: Denoising step count T, generation latents count N, denoising network $\epsilon_\theta(\cdot)$, and sampler $\Phi(\cdot)$
Input: $\mathcal{Q} = \{\mathbf{z}^1_{\tau_1}; \ldots; \mathbf{z}^T_{\tau_T}\}$, $t = \underbrace{\{\tau_0; \ldots; \tau_T\}}_{T}$
Output: $\mathcal{Q}_{\text{gen}} = \{\mathbf{z}^1_{\tau_0}; \ldots; \mathbf{z}^N_{\tau_0}\}$
$\mathcal{Q}_{\text{gen}} \leftarrow \{\}$
 # Frame-level autoregressive generation
for $i = 1$ **to** N **do**
  $Q \leftarrow \Phi(Q, t; \epsilon_\theta)$
  $\mathbf{z}^i_{\tau_0} \leftarrow \mathcal{Q}$.dequeue()
  $\mathcal{Q}_{\text{gen}}$.enqueue($\mathbf{z}^i_{\tau_0}$)
  $\mathbf{z}^T_{\tau_T} \sim \mathcal{N}(0, \mathbf{I})$
  $\mathcal{Q}$.enqueue($\mathbf{z}^T_{\tau_T}$)
end for
Return $\mathcal{Q}_{\text{gen}}$

Algorithm 3 One-step inference over the latents queue in FIFO-Diffusion

Input: Denoising network $\epsilon_\theta(\cdot)$, initial latents count f_0, sliding window stride l_{stride}, and initial sampler $\phi(\cdot)$
Input: $\mathcal{Q} = \{\mathbf{z}^1_{\tau_1}; \ldots; \mathbf{z}^T_{\tau_T}\}$, $t = \underbrace{\{\tau_0; \ldots; \tau_T\}}_{T}$
Output: $\mathcal{Q}_{\text{gen}} = \{\mathbf{z}^1_{\tau_0}; \ldots; \mathbf{z}^T_{\tau_{T-1}}\}$
 # Queue length equals denoising steps T.
$l_{\text{stride}} \leftarrow [f_0/2]$
$n_{\text{iter}} \leftarrow \lceil (T - f_0)/l_{\text{stride}} \rceil + 1$
for $i = 1$ **to** n_{iter} **do**
  if $i < n_{\text{iter}}$ **then**
    $s_{\text{index}} \leftarrow (i-1) \times l_{\text{stride}} + 1$
    $e_{\text{index}} \leftarrow s_{\text{index}} + f_0$
    $Q_{\text{temp}} \leftarrow \phi(Q[s_{\text{index}} : e_{\text{index}}], t[s_{\text{index}} : e_{\text{index}}]; \epsilon_\theta)$
    for $j = 1$ **to** l_{stride} **do**
      $Q[s_{\text{index}} + j] \leftarrow Q_{\text{temp}}[j]$
    end for
  else
    $s_{\text{index}} \leftarrow T - f_0$
    $e_{\text{index}} \leftarrow T$
    $Q_{\text{temp}} \leftarrow \phi(Q[s_{\text{index}} : e_{\text{index}}], t[s_{\text{index}} : e_{\text{index}}]; \epsilon_\theta)$
    for $j = 1$ **to** f_0 **do**
      $Q[s_{\text{index}} + j] \leftarrow Q_{\text{temp}}[j]$
    end for
  end if
end for
$\mathcal{Q}_{\text{gen}} \leftarrow \mathcal{Q}$
Return $\mathcal{Q}_{\text{gen}}$

A.1.2. Two-Stage Training-Inference Alignment Mechanism.

As discussed in Sec. 3.2, our Two-Stage Training-Inference Alignment (TTA) mechanism aims to mitigate the gap between training and inference by reducing the noise span of latents fed to the model during inference. In this section, we present the detailed implementation of TTA with pseudocode.

First, we need to initialize the latents queue. To ensure compatibility with the Zigzag Iterative Denoising in Stage 1, the adjacent L_{zig} latents are assigned the same noise level during initialization. A notable difference from the initialization in FIFO-Diffusion is that, after initializing the queue tail with clean latents, we progressively guide the noise latent using these latents to complete the initialization of the entire queue, rather than simply duplicating semantically identical frames. The detailed initialization procedure is presented in Alg. 4.

Next, we perform our designed two-stage iterative generation process, as illustrated in Alg. 5. In Stage 1 (Zigzag Iterative Denoising), after each inference on the queue, we dequeue the L_{zig} partially denoised latents from the head of the queue and save them, while L_{zig} Gaussian noise latents are enqueued at the tail. After N latents have been dequeued, Stage 2 performs the remaining denoising steps on these latents with identical noise levels.

Algorithm 4 Initial latents construction in MIGA

Input: Denoising step count T, initial latents count f_0, zigzag width L_{zig}, denoising network $\epsilon_\theta(\cdot)$, and sampler $\Phi(\cdot)$
Input: initial clean latents $q_{\text{clean}} = \{\mathbf{z}^1_{\tau_0}; \dots; \mathbf{z}^{f_0}_{\tau_0}\}$, time steps $t_{\text{clean}} = \underbrace{\{\tau_0; \dots; \tau_0\}}_{f_0}$
Output: $\mathcal{Q}_{\text{init}} = \{\underbrace{\mathbf{z}^1_{\tau_e}, \cdots, \mathbf{z}^{L_{\text{zig}}}_{\tau_e}}_{L_{\text{zig}}}, \underbrace{\mathbf{z}^{L_{\text{zig}}+1}_{\tau_{e+1}}, \cdots, \mathbf{z}^{2L_{\text{zig}}}_{\tau_{e+1}}}_{L_{\text{zig}}}, \cdots, \underbrace{\mathbf{z}^{L-L_{\text{zig}}+1}_{\tau_T}, \cdots, \mathbf{z}^{L}_{\tau_T}}_{L_{\text{zig}}}\}$,
$t_{\text{init}} = \{\underbrace{\tau_e, \cdots, \tau_e}_{L_{\text{zig}}}, \underbrace{\tau_{e+1}, \cdots, \tau_{e+1}}_{L_{\text{zig}}}, \cdots, \underbrace{\tau_T, \cdots, \tau_T}_{L_{\text{zig}}}\}$
$\mathcal{Q}_{\text{init}} \leftarrow \{\}, \quad t_{\text{init}} \leftarrow \{\}$
 # Manually add noise to the initial latents.
$n_{\text{zig}} \leftarrow \lceil f_0 / L_{\text{zig}} \rceil$
for $i = 1$ **to** f_0 **do**
  $t_{\text{index}} \leftarrow T - n_{\text{zig}} + \lceil (i-1)/L_{\text{zig}} \rceil$
  $\mathbf{z}^i_{t_{\text{index}}} \leftarrow \Phi.\text{add_noise}(q_{\text{clean}}[i], t_{\text{clean}}[i], \tau_{t_{\text{index}}})$
  $\mathcal{Q}_{\text{init}}.\text{enqueue}(\mathbf{z}^i_{\tau_{t_{\text{index}}}})$
  $t_{\text{init}}.\text{enqueue}(\tau_{t_{\text{index}}})$
end for
 # Progressively guide the generation of subsequent latents using existing latents.
for $i = 1$ **to** $T - n_{\text{zig}} - e$ **do**
  for $j = 1$ **to** L_{zig} **do**
    $\mathbf{z}^T_{\tau_T} \sim \mathcal{N}(0, \mathbf{I})$
    $\mathcal{Q}_{\text{init}}.\text{enqueue}(\mathbf{z}^T_{\tau_T})$
    $t_{\text{init}}.\text{enqueue}(\tau_{t_T})$
  end for
  $\mathcal{Q}_{\text{init}} \leftarrow \Phi(\mathcal{Q}_{\text{init}}, t_{\text{init}}; \epsilon_\theta)$
  for $j = 1$ **to** $\text{len}(t_{\text{init}})$ **do**
    $t_{\text{init}}[j] \leftarrow t_{\text{init}}[j] - 1$
  end for
end for
Return $\mathcal{Q}_{\text{init}}, \quad t_{\text{init}}$

A.1.3. Dual Consistency Enhancement Mechanism.

The modeling motivation and core concept of our Dual Consistency Enhancement (DCE) mechanisms are introduced in Sec. 3.3. In this section, we present the implementation details in the form of pseudocode. The pseudocode implementation can be understood with reference to the framework diagram shown in Fig. A1.

Algorithm 5 Two-stage training-inference alignment mechanism in MIGA

Input: $\mathcal{Q} = \{\underbrace{\mathbf{z}^1_{\tau_e}, \cdots, \mathbf{z}^{L_{\text{zig}}}_{\tau_e}}_{L_{\text{zig}}}, \underbrace{\mathbf{z}^{L_{\text{zig}}+1}_{\tau_{e+1}}, \cdots, \mathbf{z}^{2L_{\text{zig}}}_{\tau_{e+1}}}_{L_{\text{zig}}}, \cdots, \underbrace{\mathbf{z}^{L-L_{\text{zig}}+1}_{\tau_T}, \cdots, \mathbf{z}^{L}_{\tau_T}}_{L_{\text{zig}}}\}$
$t = \{\underbrace{\tau_e, \cdots, \tau_e}_{L_{\text{zig}}}, \underbrace{\tau_{e+1}, \cdots, \tau_{e+1}}_{L_{\text{zig}}}, \cdots, \underbrace{\tau_T, \cdots, \tau_T}_{L_{\text{zig}}}\}$
Output: $\mathcal{Q}_{\text{gen}} = \{\mathbf{z}^1_{\tau_0}; \ldots; \mathbf{z}^N_{\tau_0}\}$
$\mathcal{Q}_{\text{gen}} \leftarrow \{\}$
 # Stage 1: Zigzag iterative denoising.
$n_{\text{iter}} \leftarrow \lceil N/L_{\text{zig}} \rceil$
for $i = 1$ **to** n_{iter} **do**
  $Q \leftarrow \Phi(Q, t; \epsilon_\theta)$
  for $j = 1$ **to** L_{zig} **do**
    $\mathbf{z}^i_{\tau_0} \leftarrow \mathcal{Q}.\text{dequeue}()$
    $\mathcal{Q}_{\text{gen}}.\text{enqueue}(\mathbf{z}^i_{\tau_0})$
    $\mathbf{z}^T_{\tau_T} \sim \mathcal{N}(0, \mathbf{I})$
    $\mathcal{Q}.\text{enqueue}(\mathbf{z}^T_{\tau_T})$
  end for
end for
 # Stage 2: Denoising at a unified noise level.
$t \leftarrow \{\}$
for $i = 1$ **to** $\text{len}(\mathcal{Q}_{\text{gen}})$ **do**
  $t.\text{enqueue}(\tau_{e-1})$
end for
for $i = 1$ **to** $e - 1$ **do**
  $\mathcal{Q}_{\text{gen}} \leftarrow \Phi(\mathcal{Q}_{\text{gen}}, t; \epsilon_\theta)$
  for $j = 1$ **to** $\text{len}(t)$ **do**
    $t[j] \leftarrow t[j] - 1$
  end for
end for
Return $\mathcal{Q}_{\text{gen}}, \quad t$

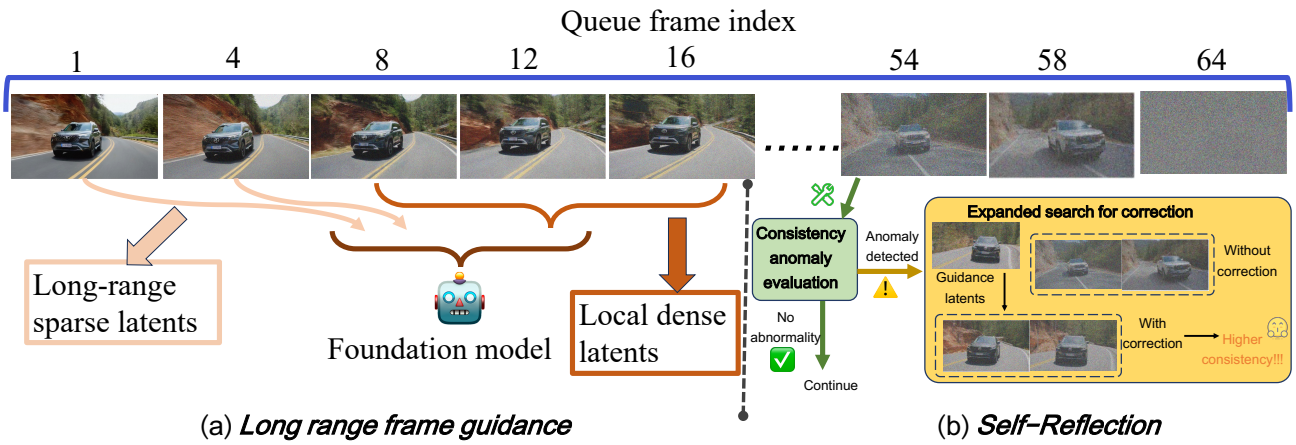


Figure A1. **Framework of our Dual Consistency Enhancement (DCE) mechanism, which consists of Long-Range Frame Guidance (a) and Self-Reflection (b) approaches. (a)** Long-Range Frame Guidance enables the foundation model to process long-range sparse latents and local dense latents simultaneously as input when handling local segments in a sliding manner. **(b)** The Self-Reflection approach starts from high-noise latents at the end of the queue and first performs consistency anomaly evaluation. When anomalies are detected (*e.g.*, the car's color changes to white whereas it was black in previous frames), expanded search for correction is conducted on these tail latents. After correction, the updated latents achieve higher consistency.

First, our Self-Reflection approach focuses on early high-noise latents along the queue dimension, performing consistency evaluation to promptly correct the latent anomalies. In conjunction with the previously described TTA iterative generation process, Self-Reflection is applied before each queue inference (*i.e.*, $Q \leftarrow \Phi(Q, t; \epsilon_\theta)$, see line 8 in Alg. 5). As shown in Alg. 6, we provide a detailed description of the implementation of the Self-Reflection approach. Note that the implementation of the Self-Reflection approach in Sec. 3.3 assumes $L_{\text{zig}} = 1$. A more general version is provided in Alg. 6.

Furthermore, the Long-Range Frame Guidance approach targets low-noise latents at the queue head to facilitate interactions among distant latents (Feng et al., 2025c), thereby improving video consistency. It is integrated into each queue inference step within the TTA iterative generation process (*i.e.*, $Q \leftarrow \Phi(Q, t; \epsilon_\theta)$, see line 8 in Alg. 5), and its implementation is detailed in Alg. 7.

A.2. Analysis of Framework Unification

In Sec. 3 and Sec. A.1, we present the methodology and pseudocode implementations of the proposed TTA and DCE mechanisms, respectively. It is important to emphasize that TTA and DCE are not independent modules. In this subsection, we clarify their interdependence and integrated design within the adopted frame-level autoregressive generation paradigm.

Frame-level Autoregressive Framework. Fig. 2 (a) illustrates the autoregressive generation process of this framework.

- At the core of this process is the maintenance of a structured queue of noisy frames, with noise intensity increasing along the temporal axis. At each inference step, the noise levels of all frames are simultaneously reduced, allowing the clean frame to be popped from the queue head. By repeating this procedure iteratively, frame-level autoregressive generation is achieved.
- Furthermore, during each inference operation on the queue, since foundation models can only process a limited number of frames at a time, a sliding window is employed to sequentially refine all noisy frames in the queue.

Unification of MIGA Mechanisms. Corresponding to the overall queue-level inference process and the local frame-level sliding-window denoising within each queue, our core TTA and DCE mechanisms are specifically designed to optimize these two aspects.

- The TTA mechanism divides the queue-level inference into two stages, each characterized by different noise intensity distributions across the frames. Through coordinated operation, TTA reduces the noise span fed to the model at each

Algorithm 6 Self-reflection approach in MIGA

Input: Judgment index f_{judg}, reference latents count f_{ref}, evaluation latents count f_{eval}, adjustment threshold δ_{adju}, guidance latents count f_{guid}, zigzag width L_{zig}, consistency score from the previous step C^0_{score}, denoising network $\epsilon_\theta(\cdot)$, and sampler $\Phi(\cdot)$
Input: $\mathcal{Q} = \{\mathbf{z}^1; \ldots; \mathbf{z}^L\}$, $t = \underbrace{\{\tau^1; \ldots; \tau^L\}}_{L}$ **# Without loss of generality, we do not distinguish the noise step of each latent.**
Output: $\mathcal{Q}_{\text{adju}} = \{\mathbf{z}^1; \ldots; \mathbf{z}^L\}, C^0_{\text{score}}$
$\mathcal{Q}_{\text{adju}} \leftarrow \{\}$
 # Evaluation
$q_{\text{ref}} \leftarrow \mathcal{Q}[f_{\text{judg}} - f_{\text{ref}} : f_{\text{judg}} - 1]$
$q_{\text{eval}} \leftarrow \mathcal{Q}[f_{\text{judg}} : f_{\text{judg}} + f_{\text{eval}} - 1]$
$C_{\text{score}} \leftarrow \text{cosine_similarity}(q_{\text{ref}}, q_{\text{eval}})$ **# See Eq. 5,Eq. 6 and Eq. 7**
if $C^0_{\text{score}} - C_{\text{score}} \leq \delta_{\text{adju}}$ **then**
  # Do not perform extended sampling.
  $\mathcal{Q}_{\text{adju}} \leftarrow \mathcal{Q}$
  $C^0_{\text{score}} \leftarrow C_{\text{score}}$
else
  # Perform extended sampling.
  $q_{\text{guid}} \leftarrow \mathcal{Q}[f_{\text{judg}} - f_{\text{guid}} : f_{\text{guid}} - 1]$
  $t_{\text{guid}} \leftarrow t[f_{\text{judg}} - f_{\text{guid}} : f_{\text{guid}} - 1]$
  $\mathcal{Q}_{\text{sample}} \leftarrow \{\}$
  for $k = 1$ **to** n_{samp} **do**
    $\mathcal{Q}_{\text{temp}} \leftarrow q_{\text{guid}}$
    $t_{\text{temp}} \leftarrow t_{\text{guid}}$
    $n_{\text{iter}} \leftarrow \lceil (L - f_{\text{judg}} + 1)/L_{\text{zig}} \rceil$
    for $i = 1$ **to** n_{iter} **do**
      $\mathcal{Q}_{\text{temp}} \leftarrow \Phi(\mathcal{Q}_{\text{temp}}, t_{\text{temp}}; \epsilon_\theta)$
      for $j = 1$ **to** L_{zig} **do**
        $\mathbf{z}^T_{\tau_T} \sim \mathcal{N}(0, \mathbf{I})$
        $\mathcal{Q}_{\text{temp}}.\text{enqueue}(\mathbf{z}^T_{\tau_T})$
        $t_{\text{temp}}.\text{enqueue}(\tau_T)$
      end for
    end for
    $\mathcal{Q}_{\text{sample}}.\text{enqueue}(\mathcal{Q}_{\text{temp}})$
  end for
  # Determine whether correction is needed
  $C_{\text{samp}} \leftarrow \{\}$
  for $k = 1$ **to** n_{samp} **do**
    $q_{\text{ref}} \leftarrow \mathcal{Q}[f_{\text{guid}} - f_{\text{ref}} : f_{\text{judg}} - 1]$
    $q_{\text{eval}} \leftarrow \mathcal{Q}[f_{\text{guid}} : f_{\text{guid}} + f_{\text{eval}} - 1]$
    $C^k \leftarrow \text{cosine_similarity}(q_{\text{ref}}, q_{\text{eval}})$
    $C_{\text{samp}}.\text{enqueue}(C^k)$
  end for
  $\max_{\text{index}}, C^{\max} \leftarrow \max(C_{\text{samp}})$
  if $C^{\max} > C_{\text{score}}$ **then**
    # Perform correction.
    $\mathcal{Q}[f_{\text{judg}} :] \leftarrow \mathcal{Q}_{\text{sample}}[\max_{\text{index}}][f_{\text{ref}} :]$
    $C^0_{\text{score}} \leftarrow C^{\max}$
  else
    # Do not perform correction.
    $\mathcal{Q}_{\text{adju}} \leftarrow \mathcal{Q}$
    $C^0_{\text{score}} \leftarrow C_{\text{score}}$
  end if
end if
Return $\mathcal{Q}_{\text{adju}}, \quad C^0_{\text{score}}$

Algorithm 7 One-step inference over the latents queue incorporates our long-range frame guidance

Input: Denoising network $\epsilon_\theta(\cdot)$, initial latents count f_0, sliding window stride l_{stride}, zigzag width L_{zig}, long-range guidance latents count m_{guid}, and initial sampler $\phi(\cdot)$
Input: $\mathcal{Q} = \{\mathbf{z}^1; \ldots; \mathbf{z}^L\}$, $t = \underbrace{\{\tau^1; \ldots; \tau^L\}}_{L}$ **# Without loss of generality, we do not distinguish the noise step of each latent.**
Output: $\mathcal{Q}_{\text{gen}} = \{\mathbf{z'}^1; \ldots; \mathbf{z'}^L\}$ **# $\mathbf{z'}^i$ represents $\mathbf{z}^i$ after one denoising step.**
 # Queue length equals denoising steps T.
$l_{\text{stride}} \leftarrow [f_0/2]$
$n_{\text{iter}} \leftarrow \lceil (T - f_0 + m_{\text{guid}})/l_{\text{stride}} \rceil + 1$
for $i = 1$ **to** n_{iter} **do**
  if $i < n_{\text{iter}}$ **then**
    $s_{\text{index}} \leftarrow (i-1) \times l_{\text{stride}}$
    if $s_{\text{index}} \leq n_{\text{iter}}$ **then**
      $e_{\text{index}} \leftarrow s_{\text{index}} + f_0$
      $\mathcal{Q}_{\text{temp}} \leftarrow \phi(\mathcal{Q}[s_{\text{index}} : e_{\text{index}}], t[s_{\text{index}} : e_{\text{index}}]; \epsilon_\theta)$
      for $j = 1$ **to** l_{stride} **do**
        $\mathcal{Q}[s_{\text{index}} + j] \leftarrow \mathcal{Q}_{\text{temp}}[j]$
      end for
    else
      $s_{\text{range}} \leftarrow \min(m_{\text{guid}} L_{\text{zig}}, s_{\text{index}} - 1)$
      $s^0_{\text{list}} \leftarrow \text{uniform_sample}(\text{range}(s_{\text{index}} - s_{\text{range}}, s_{\text{index}} - 1), m_{\text{guid}})$ **# Uniformly sample m_{guid} indices from $(s_{\text{index}} - s_{\text{range}}, s_{\text{index}} - 1)$.**
      $\mathcal{Q}_{\text{input}} \leftarrow \{\}, \quad t_{\text{input}} \leftarrow \{\}$
      for $j = 1$ **to** m_{guid} **do**
        $\mathcal{Q}_{\text{input}}.\text{enqueue}(\mathcal{Q}[s^0_{\text{list}}[j]])$
        $t_{\text{input}}.\text{enqueue}(t[s^0_{\text{list}}[j]])$
      end for
      $e_{\text{index}} \leftarrow s_{\text{index}} + f_0 - m_{\text{guid}}$
      $\mathcal{Q}_{\text{input}} \leftarrow \text{concat}([\mathcal{Q}_{\text{input}}; \mathcal{Q}[s_{\text{index}} : e_{\text{index}}]])$
      $t_{\text{input}} \leftarrow \text{concat}([t_{\text{input}}; t[s_{\text{index}} : e_{\text{index}}]])$
      $\mathcal{Q}_{\text{temp}} \leftarrow \phi(\mathcal{Q}[s_{\text{index}} : e_{\text{index}}], t[s_{\text{index}} : e_{\text{index}}]; \epsilon_\theta)$
      for $j = 1$ **to** l_{stride} **do**
        $\mathcal{Q}[s_{\text{index}} + j] \leftarrow \mathcal{Q}_{\text{temp}}[m_{\text{guid}} + j]$
      end for
    end if
  else
    $s_{\text{index}} \leftarrow T - f_0 + m_{\text{guid}}$
    $s_{\text{range}} \leftarrow \min(m_{\text{guid}} L_{\text{zig}}, s_{\text{index}} - 1)$
    $s^0_{\text{list}} \leftarrow \text{uniform_sample}(\text{range}(s_{\text{index}} - s_{\text{range}}, s_{\text{index}} - 1), m_{\text{guid}})$
    $\mathcal{Q}_{\text{input}} \leftarrow \{\}, \quad t_{\text{input}} \leftarrow \{\}$
    for $j = 1$ **to** m_{guid} **do**
      $\mathcal{Q}_{\text{input}}.\text{enqueue}(\mathcal{Q}[s^0_{\text{list}}[j]])$
      $t_{\text{input}}.\text{enqueue}(t[s^0_{\text{list}}[j]])$
    end for
    $e_{\text{index}} \leftarrow s_{\text{index}} + f_0 - m_{\text{guid}}$
    $\mathcal{Q}_{\text{input}} \leftarrow \text{concat}([\mathcal{Q}_{\text{input}}; \mathcal{Q}[s_{\text{index}} : e_{\text{index}}]])$
    $t_{\text{input}} \leftarrow \text{concat}([t_{\text{input}}; t[s_{\text{index}} : e_{\text{index}}]])$
    $\mathcal{Q}_{\text{temp}} \leftarrow \phi(\mathcal{Q}[s_{\text{index}} : e_{\text{index}}], t[s_{\text{index}} : e_{\text{index}}]; \epsilon_\theta)$
    for $j = 1$ **to** l_{stride} **do**
      $\mathcal{Q}[s_{\text{index}} + j] \leftarrow \mathcal{Q}_{\text{temp}}[m_{\text{guid}} + j]$
    end for
  end if
end for
$\mathcal{Q}_{\text{gen}} \leftarrow \mathcal{Q}$
Return $\mathcal{Q}_{\text{gen}}$

step, effectively transferring the short-term generative capabilities of foundation models to the long-video generation scenario.

- The DCE mechanism focuses on denoising within each queue iteration by employing long-range guidance from low-noise frames and timely self-reflection correction for high-noise early frames. This mechanism effectively enhances consistency in long video generation.

As depicted in Fig. 2 (b), TTA primarily focuses on the vertical dimension (queue-level inference), while DCE operates along the horizontal dimension (frame-level denoising within each queue; see also Fig. A1 for a detailed illustration). Together, these two mechanisms form a unified and coherent generative framework that enables the consistent and high-quality generation of long videos. Alg. 7 illustrates how the two mechanisms interact and operate jointly.

A.3. Multi-prompt Conditional Generation

As discussed in Sec. 4.1, the flexibility of the frame-level autoregressive framework enables multi-text control by providing different text conditions to the latents at various temporal positions. Specifically, when performing inference over the queue, the expression involving the text conditioning c is given by (for clarity, c is omitted in Eq. 2):

$$\{\mathbf{z}^1_{\tau_0}, \dots, \mathbf{z}^T_{\tau_{T-1}}\} = \Phi(\{\mathbf{z}^1_{\tau_1}, \dots, \mathbf{z}^T_{\tau_T}\}, \{\tau_1, \dots, \tau_T\}, c; \varepsilon_\theta). \tag{A2}$$

When there is only a single text condition, c is a constant (*i.e.*, the feature of a single text prompt). However, with multiple text conditions, as the foundation model iterates over the queue with a sliding window, latents at different positions are guided by different c. For the case of n_{prom} text prompts, *i.e.*, $c = \{c_i\}_{i=1}^{n_{\text{prom}}}$, each prompt sequentially controls the generation of N_{prom} frames (resulting in $N = n_{\text{prom}} N_{\text{prom}}$ clean latents for the final video). Formally, suppose the foundation model is at position l in the queue (*i.e.*, s_{index} in Alg. 3 and Alg. 7), and the number of dequeued clean latents is n_{deq}. Then, the input text condition for the model is:

$$c_{\text{in}} = c\left[\left\lceil \frac{l + n_{\text{deq}}}{N_{\text{prom}}} \right\rceil\right]. \tag{A3}$$

As shown in Fig. A2, we present a case generated by our Wan2.1-based MIGA that contains three prompts.

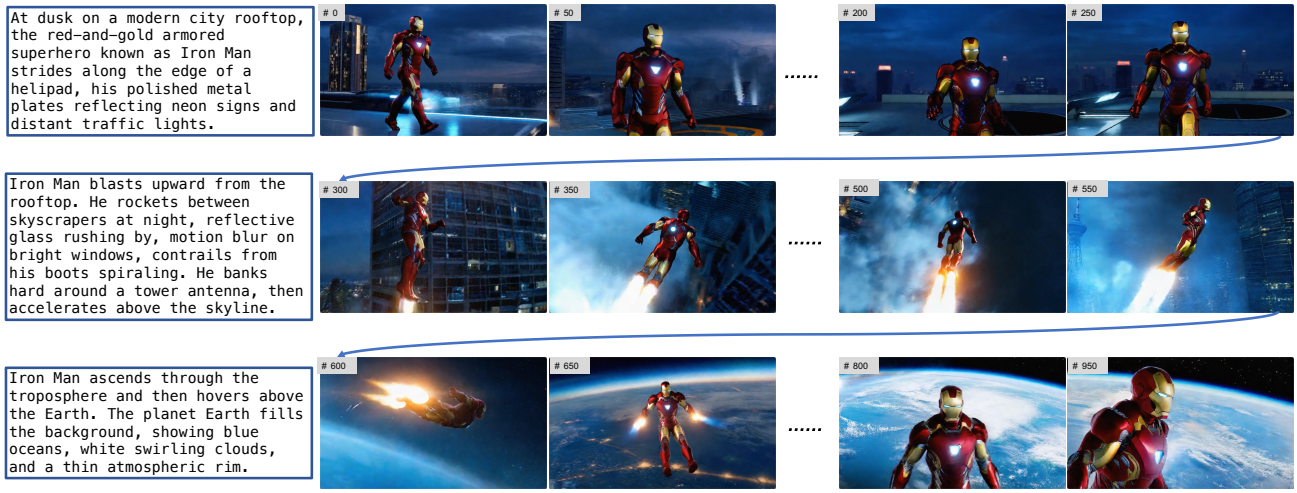


Figure A2. **Illustration of a multi-prompt controlled generation case produced by our Wan2.1-based MIGA.**

A.4. Implementation on Different Foundation Models

VideoCrafter2-Based MIGA. As an early and representative text-to-video foundation model, VideoCrafter2 (Chen et al., 2023; 2024b) is widely used as the backbone for existing train-free long video generation methods (Qiu et al., 2023; Tan et al., 2025; Lu et al., 2024; Kim et al., 2024; Yang et al., 2025a). Following its original denoising inference code, our main

modification to equip it with frame-level autoregressive generation is to adjust its noise prediction model $\epsilon_\theta(\cdot)$ and sampler (*i.e.*,DDIM (Song et al., 2020)) $\phi(\cdot)$ in the noise prediction and denoising process. Specifically, the original $\epsilon_\theta(\cdot)$ receives latents with identical noise levels each time, and $\phi(\cdot)$ applies the same denoising operation to all latents. Our key change is that $\epsilon_\theta(\cdot)$ now receives latents with different noise levels in each inference step, where the noise level is determined by the frame index. During noise prediction, each frame's latents interact with their respective noise level conditions. In the denoising stage, $\phi(\cdot)$ also processes latents for different frames separately, conditioning on their noise levels.

Wan2.1-Based MIGA. With the continuous evolution of text-to-video foundation models, train-free long video generation frameworks should also be adapted to these newer models. To this end, we migrate MIGA to the latest available model, Wan2.1 (Wan et al., 2025). Similar to the VideoCrafter2-based MIGA, the core modifications involve adjusting the noise prediction model $\epsilon_\theta(\cdot)$ and the sampler $\phi(\cdot)$ in the noise prediction and denoising process. The key difference is that Wan2.1 employs UniPC (Zhao et al., 2023) as its default sampler, which requires higher-order computations. Consequently, both $\epsilon_\theta(\cdot)$ and $\phi(\cdot)$ need to store and utilize information from previous steps during each operation.

Discussion on the Generalizability of Our Method. The frame-level autoregressive generation framework we adopt inherently requires models to handle latents with noise levels varying across frames. Our MIGA can be migrated to VideoCrafter2 and Wan2.1 because both their noise conditions and text conditions interact with latents via cross-attention. Specifically, latents can incorporate noise timestep conditions by distinguishing frame indices, while all latents are treated as a whole for text conditioning. However, We observe that this frame-level autoregressive generation framework is difficult to apply to certain foundation models (Kong et al., 2024; Yang et al., 2024) based on the MMDiT architecture (Esser et al., 2024). The main reason is that these models concatenate text and video features, and jointly interact with the noise timestep condition. To guide latents of different frames with distinct noise levels, it is necessary to introduce noise conditions with varying timesteps. However, since text features cannot be distinguished at the frame level, this noise information cannot effectively interact with the text features.. Fig. A3 illustrates our approach of injecting an intermediate timestep into the text features to enable the migration of the frame-level autoregressive generation framework to CogVideoX-5B (Yang et al., 2024), which is based on the MMDiT architecture. As shown, this results in abnormal video outputs.

Figure A3. **Illustration of a bad case resulting from migrating frame-level autoregressive generation to CogVideoX, a model based on the MMDiT architecture.**

B. Further Details on Experimental Analysis

B.1. Implementation Details of Ablation Studies

In Sec. 4.3, we conduct comprehensive ablation studies on the VideoCrafter2-based MIGA using VBench to evaluate the effectiveness of our proposed mechanism design. In this subsection, we provide detailed implementation information for each setting.

Study on Core Mechanism Designs. The core contribution of our work lies in the introduction of the novel Two-Stage Training-Inference Alignment (TTA) and Dual Consistency Enhancement (DCE) mechanisms. In Tab. 3, we use FIFO-Diffusion as the baseline setting without TTA or DCE, which serves as the baseline for our ablation study. For the TTA-only setting, we add Stage 1 zigzag iterative denoising with $L_{\text{zig}} = 4$ and Stage 2 unified noise level denoising with $\tau_e = 10$ to the baseline. For the DCE-only setting, we introduce a self-reflection method with $\delta_{\text{adju}} = 0.01$ and a long-range frame guidance (Chen et al., 2025b; Feng et al., 2025a) method with $m_{\text{guid}} = 4$ on top of the baseline. Finally, combining both mechanisms yields our final method.

Study on TTA. Our TTA mechanism consists of two stages: stage 1 employs zigzag iterative denoising, and stage 2 applies denoising at a unified noise level. In Tab. 6, FIFO-Diffusion is used as the baseline, consistent with the setting in Tab. 3. For the "+Stage 1" setting, we add only stage 1 with $L_{\text{zig}} = 4$ to the baseline. Building upon this, by incorporating stage 2 with $\tau_e = 10$, we obtain the "+Stage 2" setting.

Next, we conduct ablation studies on the two core hyperparameters in TTA, *i.e.*, L_{zig} and τ_e (where τ_e determines the number of steps in stage 2, $(e-1)$). Considering the high computational cost of long video generation, these experiments are carried out on a subset of the evaluation set, selected by randomly sampling 50% of the prompts from the full evaluation prompts. For the baseline, *i.e.*, $L_{\text{zig}} = 1$, we report the performance of FIFO-Diffusion on this subset. Then, we progressively increase L_{zig} and present the results in Tab. 4. Fig. 6 illustrates the impact of varying the number of stage 2 denoising steps on model performance, highlighting the overall score metric across different settings. The detailed results for each individual metric under these settings are presented in Tab. A1. For the baseline setting ($e = 1$), we report the performance of FIFO-Diffusion on this evaluation subset. Then, we further analyze the results for $e = 5,\ 10,\ 15,\ 25,\ 30$ based on this baseline. When e is increased to the total denoising steps (64), stage 1 is entirely omitted, and only stage 2 is performed. As illustrated in Alg. 8, N random noises are first initialized, and then the queue undergoes 64 inference steps (each executed by sliding the window across the queue with the foundation model), ultimately producing N clean latents in one pass. It is important to note that this approach completely loses the autoregressive property in generation, whereas stage 1 (when retained) preserves the autoregressive nature. Compared with performing only stage 2, stage 1—through its initialization and iterative generation (see Alg. 4 and Alg. 5)—enables the cleaner latents to implicitly guide the generation of subsequent high-noise latents, thereby enhancing the consistency of the generated videos. This connection among latents ensures video coherence, whereas only stage 2, by synchronously denoising N independent noise latents, results in weak latent correlation and poorer consistency.

A qualitative explanation of the synergy between these two stages is as follows: stage 1 is responsible for the early denoising of latents, leveraging the autoregressive mechanism to build connections among latents and maintain semantic and spatial consistency (Qiu et al., 2023). Subsequently, stage 2 completes the final denoising process. Once the overall content has been established, stage 2 matches the input conditions seen during training (*i.e.*, the noise span is 1), which contributes to improved visual details in the generated videos. As shown in Fig. 4 (a–c), stage 2 effectively suppresses visual artifacts such as noise in the output videos.

Table A1. Ablation study on steps of stage 2.

Steps	S.C.	B.C.	M.S.	T.F.	O.S.
0	94.23	94.52	97.98	96.47	95.80
5	93.58	95.33	98.19	96.68	95.95
10	93.64	95.73	98.11	96.36	95.96
15	94.05	95.37	98.23	96.74	96.10
20	94.07	95.42	98.24	96.80	96.13
25	93.54	95.58	98.30	97.09	96.13
30	93.92	95.70	98.17	96.64	96.11
64	89.91	94.45	97.16	95.47	94.25

Algorithm 8 Inference procedure with stage 2 only

Input: Denoising step count T, generation latents count N, denoising network $\epsilon_\theta(\cdot)$, and sampler $\Phi(\cdot)$
Input: $\mathcal{Q} = \{\mathbf{z}^1_{\tau_T}; \mathbf{z}^2_{\tau_T}; \ldots; \mathbf{z}^N_{\tau_T}\}$, $t = \underbrace{\{\tau_T; \tau_T; \ldots; \tau_T\}}_{N}$
Output: $\mathcal{Q}_{\text{gen}} = \{\mathbf{z}^1_{\tau_0}; \ldots; \mathbf{z}^N_{\tau_0}\}$
for $i = 1$ **to** T **do**
  $Q \leftarrow \Phi(Q, t; \epsilon_\theta)$
  for $j = 1$ **to** T **do**
    $t[j] \leftarrow t[j] - 1$
  end for
end for
$\mathcal{Q}_{\text{gen}} \leftarrow \mathcal{Q}$
Return $\mathcal{Q}_{\text{gen}}$

Study on DCE. Our DCE mechanism is composed of self-reflection and long-range frame guidance. Fig. 5 shows the impact of different adjustment thresholds δ_{adju}. The baseline (*i.e.*, $\delta_{\text{adju}} = 0.07$, where no extended search is performed) is consistent with that used in the above TTA experiments. Based on this baseline, we progressively decrease δ_{adju}, which increases the frequency of extended search. Fig. 5 focuses on the overall score (O.S.), while detailed performance on each

metric is provided in Tab. A2. Tab. 5 presents the effect of the number of long-range guiding frames m_{guid} on model performance. The same baseline as in Fig. 5 is adopted, and results under different settings are obtained by gradually increasing m_{guid}. The computational costs introduced by these two approaches are discussed in Sec. B.3.

Table A2. Ablation study on the threshold δ_{adju}.

Threshold	S.C.	B.C.	M.S.	T.F.	O.S.
0.001	95.91	96.04	98.55	97.93	97.11
0.003	95.69	96.27	98.52	97.87	97.09
0.005	95.60	95.92	98.58	97.98	97.02
0.007	95.19	95.96	98.53	97.89	96.89
0.01	95.44	96.06	98.58	97.97	97.01
0.03	94.51	95.01	98.53	97.90	96.49
0.05	93.70	94.61	98.40	97.70	96.10
0.07	94.23	94.52	97.98	96.47	95.80

B.2. Qualitative Comparison between VideoCrafter2-based MIGA and Wan2.1-based MIGA

In Sec. 4.2, we present the results of Wan-2.1-based MIGA and VideoCrafter2-based MIGA on VBench and NarrLV. As reported in Tab. 1, one noteworthy observation on the VBench results is that, contrary to intuition, MIGA built on the latest foundation model Wan-2.1 demonstrates weaker performance in subject and background consistency than MIGA based on the earlier foundation model VideoCrafter2. The main reason is that VideoCrafter2 primarily generates animation-style videos, where maintaining long-term consistency is relatively easier than in the realistic-style videos produced by Wan-2.1. To provide a clearer explanation, Fig. A4 showcases the generation results from both approaches for the same prompt, along with their VBench evaluation scores. As seen from the comparison between Fig. A4 (a) and (b), VideoCrafter2-based MIGA tends to produce more animation-like videos, while Wan-2.1-based MIGA generates content with richer texture details. Correspondingly, the former achieves higher scores in subject and background consistency, while the latter, benefiting from Wan-2.1's capability to generate coherent video content, attains better performance in motion smoothness and temporal flicker metrics. Moreover, the evaluation results on NarrLV further demonstrate Wan-2.1's strong ability to generate videos with richer narrative content.

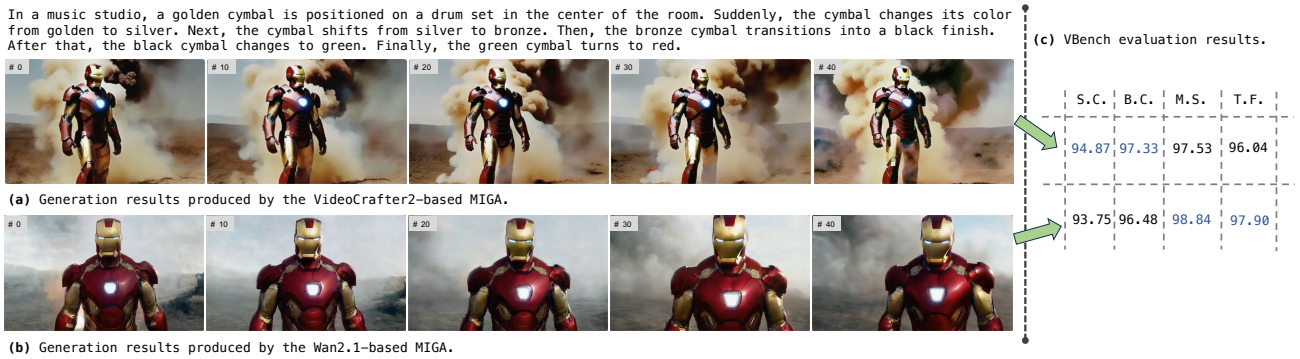


Figure A4. **Illustration of qualitative comparison between VideoCrafter2-based MIGA and Wan2.1-based MIGA (a–b), along with corresponding VBench evaluation results (c).** S.C., B.C., M.S., and T.F. denote subject consistency, background consistency, motion smoothness, and temporal flicker, respectively. The superior result for each metric is highlighted in blue.

B.3. Computational Efficiency Analysis

FIFO-Diffusion (Kim et al., 2024) demonstrates the advantages of the frame-level autoregressive generation framework in terms of memory usage and inference time. Building on this, we analyze the computational efficiency of our method. Specifically, when only the Two-Stage Training-Inference Alignment (TTA) mechanism is introduced, the computational efficiency remains identical to that of the original FIFO-Diffusion. This is because, for each latent, the required number of denoising steps is T in both approaches, resulting in equal computational cost for generating videos of the same length. As shown in Tab. 3, with comparable computational efficiency, our TTA mechanism yields a notable performance improvement (overall score increases by 2.03%), thereby demonstrating its effectiveness.

It is worth noting that Stage 2 of TTA requires all frames within a long video to be denoised to a unified noise level before performing the final denoising step. Consequently, the number of frames processed in Stage 2, *i.e.*, the maintained queue length, grows with the video length. However, since we employ a sliding-window denoising strategy, the number of frames processed by the foundation model in each window remains constant. As a result, MIGA does not incur significant additional memory overhead as the number of generated frames increases, and, like FIFO-Diffusion, it naturally supports infinite frame generation. Beyond this theoretical analysis, we further conduct a quantitative evaluation of peak memory consumption (MiB) for VideoCrafter2-based MIGA under varying generation lengths and settings. In the Tab. A3, #1 denotes the ablation variant without Stage 2. For reference, we also report the memory footprint of the foundation model (VideoCrafter2) during short-term inference, which is 9919 MiB. Values in parentheses indicate the relative memory increase compared to VideoCrafter2. The results indicate that: (i) introducing Stage 2 does not affect memory overhead across different frame counts; and (ii) memory usage increases moderately as more frames are generated due to the storage of intermediate variables, while the additional overhead remains minimal and acceptable.

Table A3. Peak memory consumption (MiB) under different generation lengths.

#	Setting	500	1000	1500	2000
1	MIGA w/o Stage 2	9929 (+0.10%)	9945 (+0.26%)	9965 (+0.46%)	9985 (+0.66%)
2	MIGA	9929 (+0.10%)	9945 (+0.26%)	9965 (+0.46%)	9985 (+0.66%)

Building upon TTA, our Dual Consistency Enhancement (DCE) mechanism introduces the additional computational cost to further improve the quality of generated videos. Specifically, the long-range frame guidance approach mainly affects the inference process for the maintained queue. Without this mechanism, as shown in Alg. 3, completing one queue inference requires the foundation model to perform $n_{\text{iter}} = \lceil (T - f_0)/l_{\text{stride}} \rceil + 1$ noise prediction steps, where $l_{\text{stride}} = [f_0/2]$. When long-range frame guidance is enabled, as shown in Alg. 7, the foundation model needs to perform $n_{\text{iter}} = \lceil (T - f_0 + m_{\text{guid}})/l_{\text{stride}} \rceil + 1$ noise prediction steps per queue inference. It can be seen that the additional number of noise prediction steps introduced by long-range frame guidance is small, *i.e.*, the extra computational overhead is negligible. Nevertheless, Tab. 5 demonstrates that long-range frame guidance yields considerable performance gains. The self-reflection approach belongs to the scope of Test-Time-Scaling (TTS) technologies, which aim to improve generation quality by increasing inference time. Here, each noise prediction by the model is treated as a basic computational unit. Specifically, the additional computational burden is mainly introduced through extended sampling. As shown in Alg. 6, in each generation process, the number of extended sampling iterations n_{adju} is determined by the threshold δ_{adju}. Once extended sampling is triggered, for each of the n_{samp} samples, the foundation model must perform $\lceil (L - f_{\text{judg}} + 1)/L_{\text{zig}} \rceil$ noise prediction steps. Thus, the total number of extra noise predictions required by the self-reflection method is:

$$n_{\text{adju}} \times n_{\text{samp}} \times (\lceil (L - f_{\text{judg}} + 1)/L_{\text{zig}} \rceil) \,. \tag{A4}$$

As an optional mechanism, the degree of TTS can be flexibly controlled by adjusting n_{adju} (via modifying δ_{adju}) and n_{samp}. As shown in Fig. 5, model performance improves as computational cost increases.

To enable a quantitative analysis, we report both computational efficiency and generation quality under different settings. Specifically, computational efficiency is measured by the average time required to generate a single frame (M_t), while performance is evaluated using the Overall Score (O.S.) on VBench. As shown in Tab. A4, without the DCE mechanism, MIGA achieves computational efficiency comparable to FIFO-Diffusion, with only a marginal increase in inference time (+0.05 s), while delivering a substantial performance improvement (+1.73 O.S.). Incorporating DCE further boosts performance, albeit at the cost of increased computational overhead.

Table A4. Comparison of computational efficiency and performance under different settings.

#	Setting	M_t (s)	O.S.
1	FIFO-Diffusion	7.48	95.02
2	MIGA w/o DCE	7.53	96.75
3	MIGA	9.16	97.82

Table A5. Comparison with trained-based long video generation methods.

Method	S.C.	B.C.	M.S.	T.F.	O.S.
CausVid	97.89	96.53	98.03	96.49	97.24
Self-Forcing	97.13	96.02	98.44	96.96	97.14
LongLive	98.00	96.76	98.74	97.34	97.71
Infinity-RoPE	98.61	97.03	98.89	97.79	98.08
Reward Forcing	96.44	95.66	98.40	96.42	96.73
MIGA (Wan2.1-Based)	96.46	95.50	98.85	98.14	97.24
MIGA (VideoCrafter2-Based)	97.66	96.99	98.60	98.03	97.82

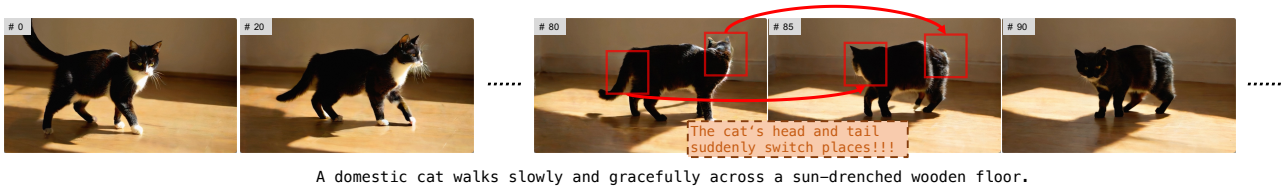


Figure A5. **Illustration of a bad case reflecting the limitations of our MIGA.**

B.4. Comparison with Training-Based Methods

Recently, some train-based long video generation models have demonstrated frame-level autoregressive generation capabilities. Although these models fall outside the scope of our work on train-free long video generation, which aims to endow foundation video generation models with long-term generation ability at minimal computational cost (Lu et al., 2024), train-based approaches instead achieve long video generation by designing specific training strategies and model architectures under a predetermined computational budget. Nevertheless, since these recent train-based models share the frame-level autoregressive inference characteristic with our MIGA framework, we provide a discussion and comparison in this subsection.

Specifically, CausVid (Yin et al., 2025) adapts a pretrained bidirectional diffusion Transformer into a frame-wise autoregressive Transformer through a specialized masking mechanism. It also reduces latency by extending Distribution Matching Distillation (Yin et al., 2024a;b) to the video domain. Following this paradigm, Self-forcing (Huang et al., 2025) mitigates exposure bias by aligning training and inference conditions. To support long-term training and enhance temporal consistency, LongLive (Yang et al., 2025b) introduces a novel streaming long-tuning strategy and a frame sink mechanism. Infinity-RoPE (Yesiltepe et al., 2025) achieves flexible and controllable long video generation by optimizing the 3D rotational position encoding from foundation models. Reward Forcing (Lu et al., 2025) dynamically adjusts sink frames and introduces the reward signals, effectively alleviating content repetition and reduced dynamics associated with sink frames. As shown in Tab. A5, we adopt the same evaluation settings as in Sec. 4.1 to compare these train-based methods. Despite not performing large-scale training, MIGA still achieves comparable performance across all metrics.

B.5. Human Evaluation Experiments

In addition to existing benchmark evaluations, we conduct a large-scale user study to further validate the effectiveness of our method. Specifically, we compare our MIGA with its improved baseline, FIFO-Diffusion. We randomly sample 48 prompts as generation conditions, producing 48 pairs of videos. Eight annotators are invited to perform pairwise comparisons along four dimensions: subject consistency, background consistency, motion smoothness, and temporal flicker. For each dimension, annotators select the better video or indicate a tie. As shown in Table A6, MIGA consistently outperforms FIFO-Diffusion across all four criteria. We note that consistency remains a central and challenging objective in video generation.

B.6. More Qualitative Results

Fig. 1 presents three long video cases generated by MIGA based on Wan2.1-1.3B. Additionally, more generation cases produced by MIGA using Wan2.1-1.3B and VideoCrafter2 are shown in Fig. A6 and Fig. A7. All these generated videos are approximately one minute in length. According to their default fps settings, Wan2.1-based MIGA produces long videos with 1000 frames, while VideoCrafter2-based MIGA generates long videos with 600 frames.

Table A6. Results of the user study comparing MIGA and FIFO-Diffusion (%).

Metric	MIGA Better	Tie	FIFO-Diffusion Better
Subject Consistency	62.23	21.88	15.89
Background Consistency	61.72	20.83	17.45
Motion Smoothness	66.14	19.79	14.06
Temporal Flicker	66.14	17.70	16.15

C. Limitations and Future Work

Our proposed MIGA provides an effective train-free approach for extending the generated length of existing foundation models. While longer video duration offers greater space for content creation, it also increases the risk of unintended model behaviors. As shown in Fig. A5, the beginning of the generated video follows the text prompt well, with a cat walking from left to right. However, after some time, the cat's head and tail suddenly switch places. This phenomenon can be regarded as a hallucination (Chu et al., 2024; Bai et al., 2024) of the video generation model, or as evidence of the lack of underlying physical knowledge (Lin et al., 2025; Bansal et al., 2024). Such issues are not only specific to long video generation tasks but also represent a major challenge for the entire field of video generation (Kang et al., 2024). In future work, we aim to incorporate additional conditioning signals beyond text instructions to enable the generation of more realistic long videos.

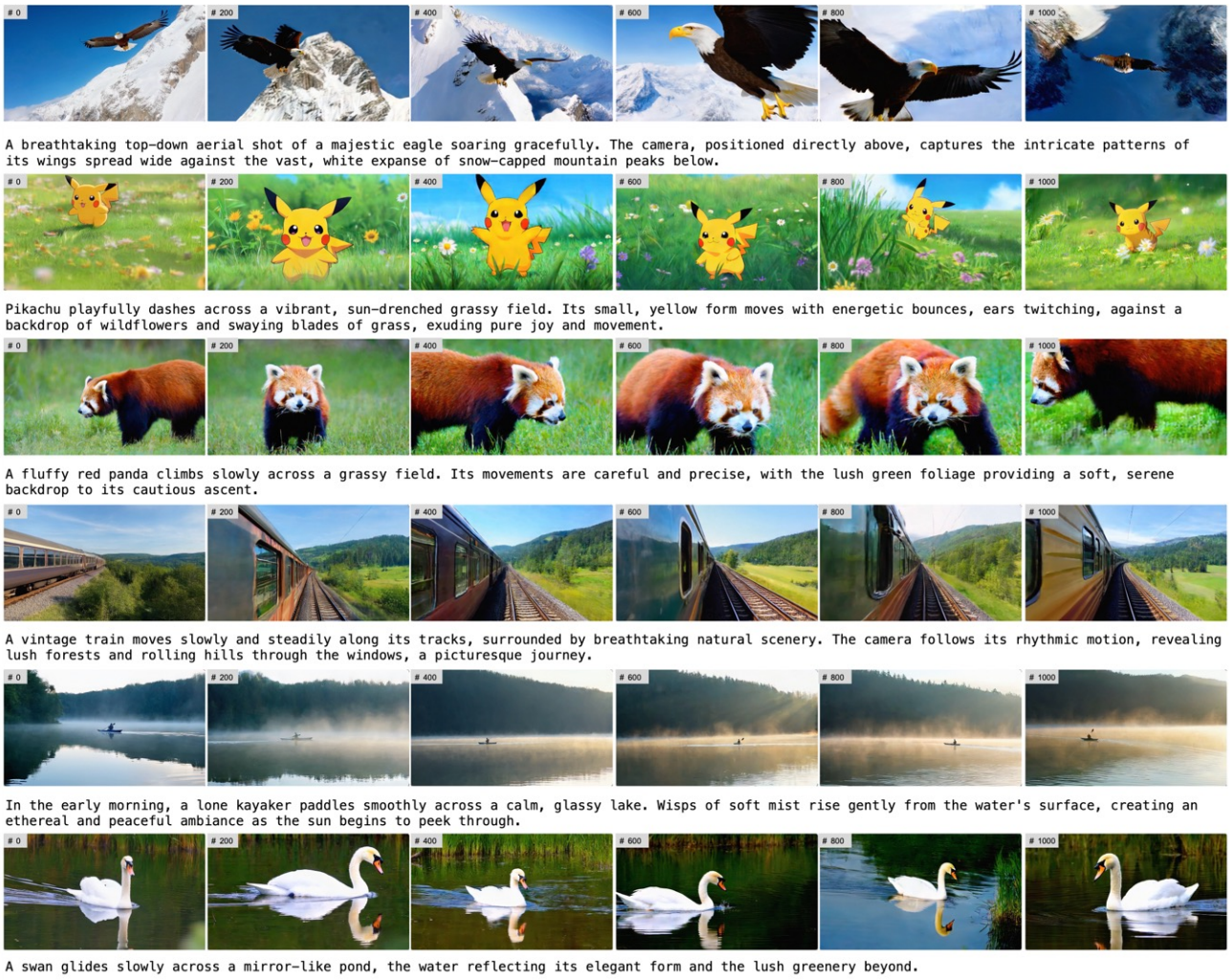


Figure A6. **Long video cases generated by Wan2.1-based MIGA.**

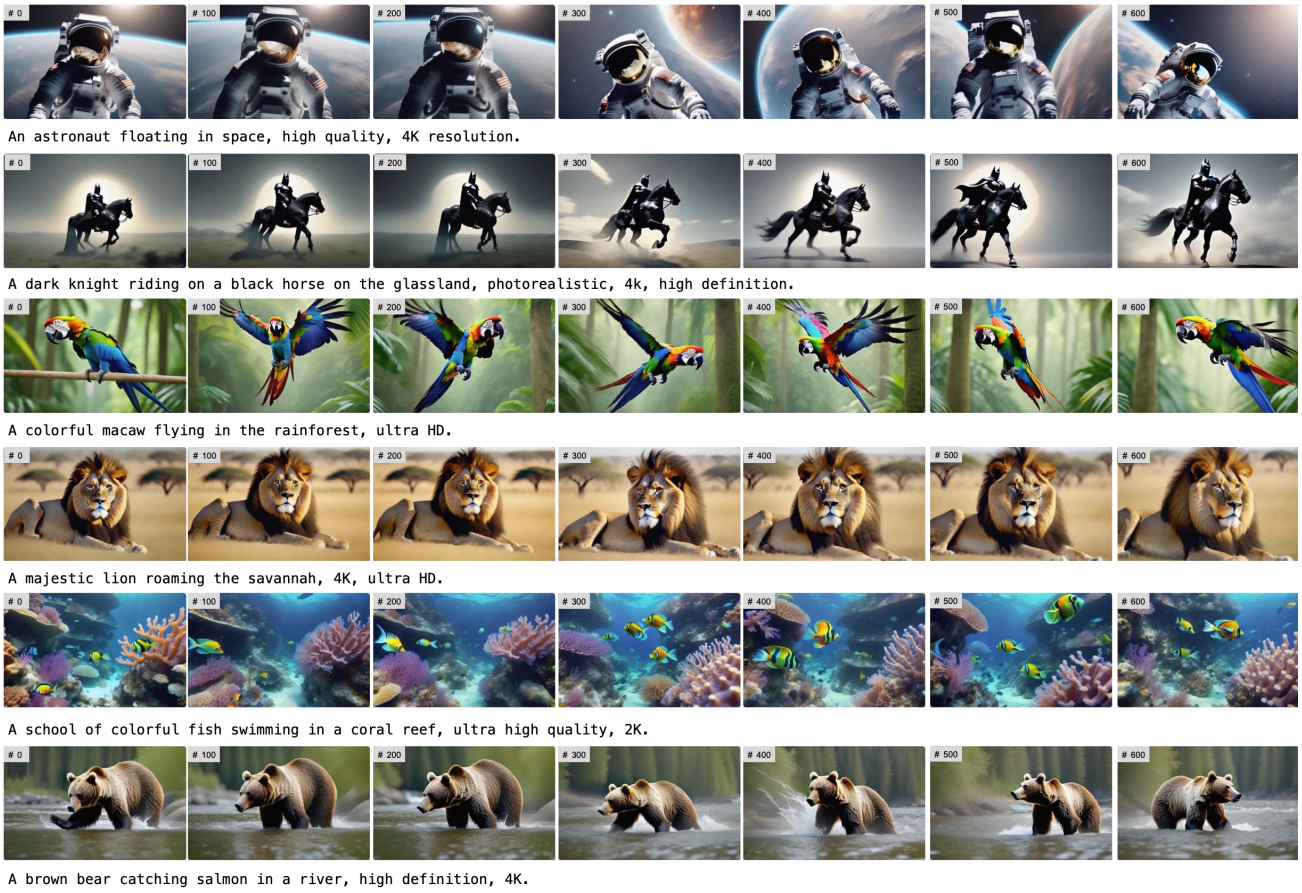


Figure A7. **Long video cases generated by VideoCrafter2-based MIGA.**