# UNIFORM DISCRETE DIFFUSION WITH METRIC PATH FOR VIDEO GENERATION

**Anonymous authors**
Paper under double-blind review

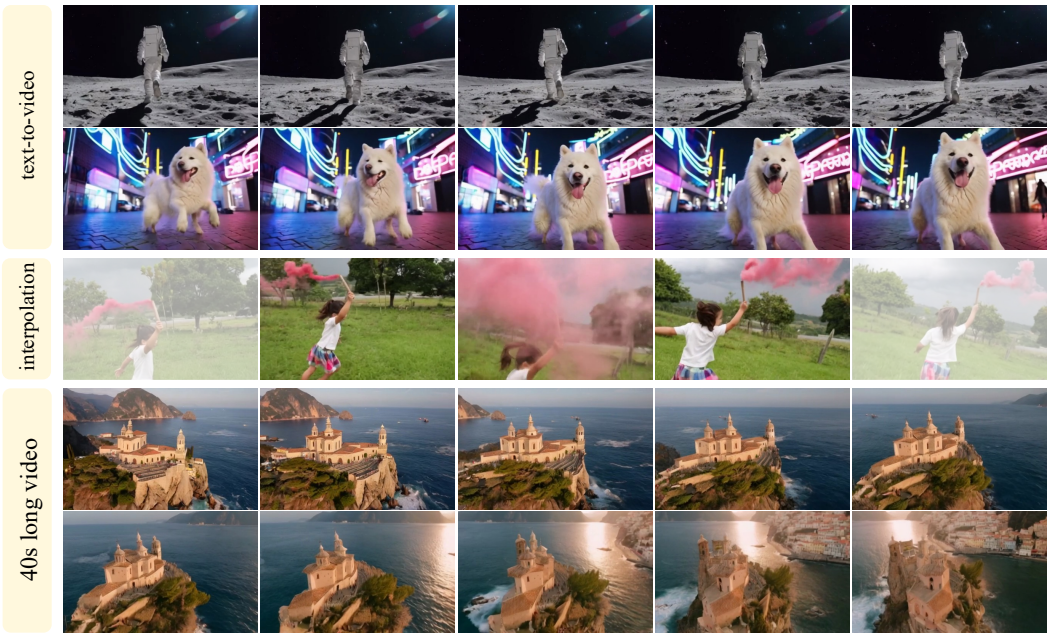


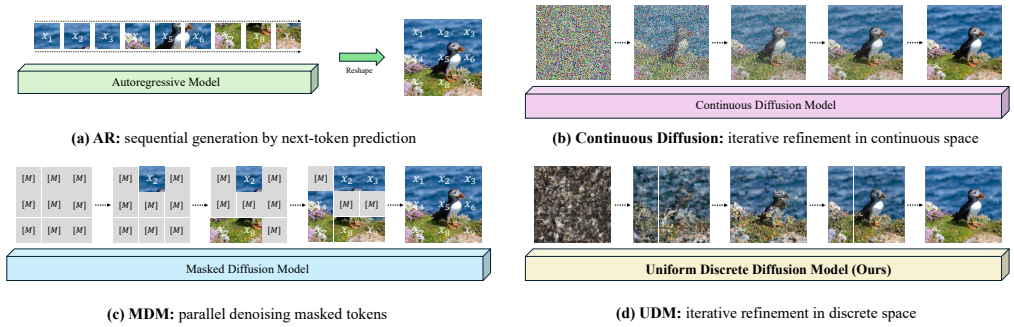

Figure 1: Visualization of **UDM** across diverse video generation tasks: text-to-video generation, video interpolation, and long video generation. These examples highlight the model's ability to synthesize coherent content, produce smooth temporal transitions, and scale to extended sequences.

## ABSTRACT

Continuous-space video generation has advanced rapidly, while discrete approaches lag behind due to error accumulation and long-context inconsistency. In this work, we revisit discrete generative modeling and present **U**niform **D**iscrete diffusion with **M**etric path (**UDM**), a simple yet powerful framework that bridges the gap with continuous methods and enables scalable video generation. At its core, UDM formulates video synthesis as iterative refinement over discrete spatio-temporal tokens. It is based on two key designs: a Linearized Metric-Path and a Resolution-dependent Timestep Shifting mechanism. This design enables UDM to scale efficiently to high-resolution image synthesis and long-duration video generation (up to 32k tokens), while requiring significantly fewer inference steps. Additionally, we introduce an asynchronous temporal fine-tuning strategy that unifies multiple tasks, including video interpolation and image-to-video synthesis, within a single model. Extensive experiments on challenging video and image generation benchmarks show that UDM consistently outperforms prior discrete methods and achieves performance comparable to state-of-the-art continuous diffusion methods.

## 1 INTRODUCTION

Continuous-space visual generation has achieved remarkable progress in both image and video synthesis (Batifol et al., 2025; Baldridge et al., 2024; Betker et al., 2023; Brooks et al., 2024; Wang




**(a) AR:** sequential generation by next-token prediction

**(b) Continuous Diffusion:** iterative refinement in continuous space

**(c) MDM:** parallel denoising masked tokens

**(d) UDM:** iterative refinement in discrete space

Figure 2: **Illustration of different image/video generation paradigms.** Discrete-space approaches such as AR and MDM adopt non-refinable local generation, where produced tokens are fixed once generated. In contrast, UDM introduces iterative global refinement, conceptually aligning discrete methods with continuous-space approaches, and substantially narrowing their performance gap.

et al., 2025a; Gao et al., 2025b; Yang et al., 2025b; Kong et al., 2024). Driven by advances in diffusion model algorithms (Ho et al., 2020; Song et al., 2021), these continuous-space methods have demonstrated strong capabilities in producing high-fidelity and visually coherent content, establishing themselves as the dominant paradigm for generative modeling.

In parallel, discrete-space text generation has become the *de facto* paradigm for large language models (Radford et al., 2018; 2019; Brown et al., 2020). Inspired by the success of LLMs, recent works have extended similar ideas to visual generation through discrete tokenization, using either next-token prediction (Sun et al., 2024a; Wang et al., 2024b; Kondratyuk et al., 2024) or masked token prediction (Chang et al., 2023; Xie et al., 2025c). However, discrete approaches still lag behind their continuous counterparts, facing challenges such as error accumulation and maintaining long-context consistency, especially in video generation. For instance, even though masked diffusion models employ bidirectional transformers, we still observe low visual quality and unnatural object motions.

In this work, we first revisit discrete generative modeling and introduce **UDM**, a powerful visual generation framework built upon **U**niform **D**iscrete diffusion models with **M**etric path. Our approach is simple: we generate videos and images by iterative refinement over discrete spatio-temporal tokens. As illustrated in Fig. 2, unlike classic autoregressive (AR) models and masked diffusion models (MDM) that adopt non-refinable local generation, where produced tokens are fixed once generated, UDM emphasizes *iterative refinement over global discrete tokens*, conceptually aligning discrete methods with continuous counterparts, and substantially narrowing their performance gap. UDM starts from categorical noise, $x_0 \sim \mathrm{Unif}([K])^D$, where each token is independently sampled from the vocabulary, and iteratively performs global refinement along a metric-guided probability path to obtain $x_1$ on the data manifold, *i.e.*, the target image or video. This iterative process enables UDM to capture the hierarchical structure of video data, from global layouts to detailed dynamics, while leveraging temporal redundancy to preserve spatio-temporal coherence.

We propose a novel metric probability path tailored for long sequences by incorporating two key components: a linearized metric path and a resolution-dependent timestep shifting mechanism. Collectively, these designs provide precise control over data perturbations, a property that is essential for maintaining stability and achieving effective training. This construction allows UDM to scale efficiently to long-sequence tasks, such as high-resolution image synthesis and long video generation, while requiring substantially fewer inference steps. Furthermore, we introduce an asynchronous timestep scheduling strategy, where timesteps are independently sampled for each frame. This asynchronous design empowers UDM to generate videos of minute-level duration and to support diverse tasks within a unified framework, including image-to-video generation and video extrapolation.

UDM attains a text-to-video score of 82.4 on VBench (Huang et al., 2024a), outperforming both discrete and continuous baselines. In image-to-video generation tasks, UDM reaches a VBench score of 86.2, on par with state-of-the-art open-source models. For text-to-image generation, UDM attains a GenEval (Ghosh et al., 2024) score of 0.8, surpassing previous discrete approaches. Furthermore, UDM exhibits strong zero-shot generalization across diverse contexts, highlighting its versatility.

Our contributions can be summarized as follows: 1) We propose Uniform Discrete diffusion with Metric path (UDM), a simple yet powerful framework that bridges the gap with continuous methods

and enables scalable video generation. 2) We highlight two key designs, linearized metric path and resolution-dependent timestep shifting, for stabilizing long-sequence training, and propose an asynchronous scheduling strategy that enables scalable video generation and unified multi-task adaptation. 3) UDM substantially pushes the envelope of discrete generation, attaining state-of-the-art results on VBench, GenEval and DPG-Bench (Hu et al., 2024).

## 2 Related Works

### 2.1 Continuous-Space Visual Generation

Continuous methods for visual generation have achieved significant progress in recent years. Early endeavors such as variational autoencoders (VAEs) (Kingma & Welling, 2013) and flow-based models (Dinh et al., 2014; 2016) exploit continuous latent spaces to model complex images, while GANs(Goodfellow et al., 2020) generate high-resolution images with strong perceptual quality via adversarial training (Brock et al., 2018; Karras et al., 2020). Diffusion models (Ho et al., 2020; Song et al., 2021), which learn to recover data by progressively denoising Gaussian noise in a continuous space, demonstrated remarkable performance in both image and video generation (Gao et al., 2025a; Batifol et al., 2025; Baldridge et al., 2024; Betker et al., 2023; Wu et al., 2025a; Brooks et al., 2024; Kong et al., 2024; Gao et al., 2025b; Wang et al., 2025a; Kuaishou, 2024; Ma et al., 2025). MAR (Li et al., 2024) employs an autoregressive framework with a diffusion head to produce continuous-valued outputs, and NOVA (Deng et al., 2025b) further extends this idea to video generation, applying autoregressive modeling to spatio-temporal sequences. UDM shares the same spirit as continuous diffusion models, performing global iterative refinement, but operates over discrete tokens.

### 2.2 Discrete-Space Visual Generation

Discrete visual generation can be broadly categorized into autoregressive and masked diffusion models, both operating on discrete visual tokens such as pixels (Kalchbrenner et al., 2017; Reed et al., 2017) or latent codes (Oord et al., 2017; Esser et al., 2021). Autoregressive models generate discrete visual tokens sequentially, with each prediction conditioned on previously generated context. This approach has been applied to both image (Sun et al., 2024a; Ramesh et al., 2021; Ding et al., 2021; Yu et al., 2022) and video synthesis (Wang et al., 2024b; Yan et al., 2021; Kondratyuk et al., 2024; Wang et al., 2024c). Although conceptually simple, this design often suffers from slow inference and pronounced error accumulation. In contrast to autoregressive methods, masked diffusion models (Gat et al., 2024; Chang et al., 2022; 2023; Yu et al., 2023) introduces the prediction of masked tokens, enabling parallel generation and improved modeling of global context. Despite these advantages, it remains challenging to apply these methods to long sequences, *e.g.* high-fidelity long-form video. FUDOKI (Wang et al., 2025b) investigates the integration of discrete flow matching (Gat et al., 2024) within native multimodal models. In this work, we adopt a uniform discrete diffusion approach, which performs iterative global refinement from categorical noise. By addressing key challenges, UDM enables both efficient inference and high-quality long-sequence generation.

## 3 Methodology

We first review the concepts of uniform discrete diffusion / discrete flow matching in Sec. 3.1, which provide the theoretical foundation for our framework. In Sec. 3.2.1-3.2.2, we introduce UDM, a simple yet powerful framework that bridges the gap between discrete and continuous approaches, enabling scalable and effective video generation.

### 3.1 Preliminary: Discrete Flow Matching

Discrete Flow Matching (DFM) (Gat et al., 2024) introduces a family of generative models designed to map data from an initial distribution, $p_0(x)$, to a final distribution, $p_1(x)$, within a discrete state space. The model utilizes a time-dependent probability path, $p_t(x)$, which interpolates between these two distributions over the interval $t \in [0, 1]$. The key idea behind DFM is to define a velocity field, $u_t$, which drives the evolution of this probability path, enabling the model to simulate a Markov process and generate new data samples.

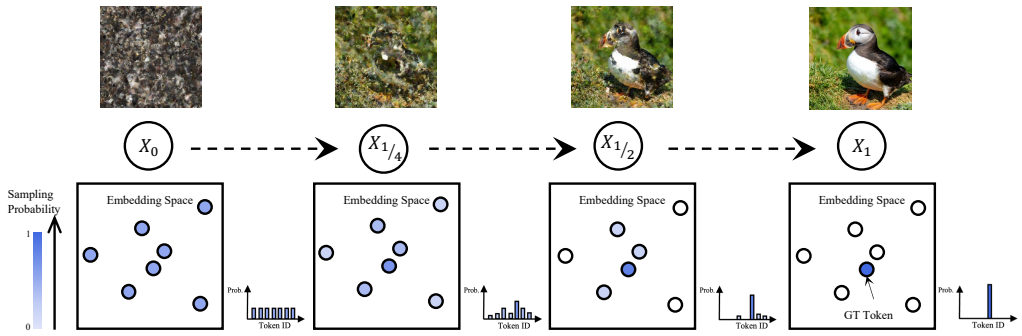



Figure 3: **A global refinement process used in UDM.** Leveraging visual token similarity, our framework captures hierarchical structures—spatial for images and spatio-temporal for videos—enabling coherent, hierarchical structure generation from global semantics to fine details.

**Probability paths.** We consider the probability path $p_t(x)$, where $t \in [0, 1]$ indexes a time-dependent probability distribution between a source distribution $p_0(x)$ and a target distribution $p_1(x)$ over $t$. Given a data distribution $q(x)$ over $x = (x^1, \ldots, x^D) \in \mathcal{S} = \mathcal{T}^D$, the probability path is defined as:

$$p_t(x) \triangleq \sum_{x_1 \in \mathcal{S}} p_t(x|x_1)q(x_1), \text{ where } p_t(x|x_1) \triangleq \prod_{i=1}^{D} p_t(x^i|x_1^i), \tag{1}$$

$p_t(x^i \mid x_1^i)$ denotes a *conditional* probability path, characterizing the evolution of the state $x^i$ given the initial state $x_1^i$.

**Probability velocities.** To generate the predefined probability path $p_t(x)$, we consider a Continuous-Time Markov Chain (CTMC), modeled as a stochastic process $X_t$. The dynamics of this CTMC are governed by a probability velocity $u_t^i$, also known as the *transition rate*. The transition rate models how the current state $x_t^i$ evolves toward the target state $x_1^i$ over time. Within this framework, each token is updated independently according to the following transition rule:

$$x_{t+h}^i \sim \delta_{x_t^i}(\cdot) + h\, u_t^i(\cdot \mid x_t^i, x_1^i), \tag{2}$$

where $u_t^i(\cdot \mid x_t^i, x_1^i)$ represents *velocity filed*, a conditional rate function that governs the flow of probability from the current state $x_t^i$ to the target state $x_1^i$ over time. Equation (2) can be interpreted as a small perturbation of the point mass $\delta_{x_t^i}$, scaled by the step size $h$, effectively modeling discrete state transitions as a continuous-time stochastic process. This velocity field is central to DFM, as it characterizes the dynamics of the probability path and is the primary quantity learned during training.

### 3.2 Uniform Discrete Diffusion with Metric Path

We present UDM, a novel framework built upon uniform discrete diffusion with metric path for image and video generation. In this section, we first introduce three key innovations: (1) a Linearized Metric-Path for structured and tractable trajectory design, (2) a Resolution-dependent Timestep Shifting mechanism to improve training stability and representation learning for long video sequences, and (3) a Frame-wise Independent Perturbation Scheduling strategy for unified long-video generation and multitask learning. After introducing these core components, we further provide the training procedure and sampling process.

#### 3.2.1 Metric Probability Path for Long Sequence Data

For data with varying sequence lengths, the degree of perturbation should be adapted during training. This requires a probability path to effectively handle sequences of different lengths, such as high-resolution images or videos. In this section, we introduce two key techniques, linearized metric path and resolution-dependent timestep shifting, to address this challenge, ensuring that the perturbation process is appropriately adjusted based on the sequence length.

**Linearized metric path.** We introduce the linearized metric path, a novel probability path derived from token embedding distances. Formally, we define the distance function $\mathrm{d} : \mathcal{T} \times \mathcal{T} \to \mathbb{R}_{\geq 0}$,

which measures the discrepancy between the codebook embeddings of generated token $x$ and the target tokens $x_1$. The distance satisfies the property $\mathrm{d}(x, x_1) = 0 \Leftrightarrow x = x_1$, ensuring a well-defined metric structure. Based on this, the probability path is defined as

$$p_t(x|x_1) = \text{softmax}\left(-\beta_t \mathrm{d}(x, x_1)\right), \tag{3}$$

where $\beta_t : [0, 1] \to \mathbb{R}_{\geq 0}$ is a monotonic scheduler function with boundary conditions $\beta_0 = 0$, $\beta_1 = \infty$. The core of linearized path lies in the functional form of $\beta_t$, which is parameterized as

$$\beta_t = c \times \left(\frac{t}{1-t}\right)^{\alpha}, t \in [0, 1), \tag{4}$$

where $c > 0$ and $\alpha > 0$ are hyperparameters that control the relationship between the sampling distance $\mathrm{d}(x_t, x_1)$ and time $t$. Specifically, the forward process samples $x_t \sim p_{t|1}(\cdot \mid x_1)$, with boundary conditions yielding a uniform distribution over codebook embeddings at $t = 0$ and a deterministic sample at $x_1$ when $t = 1$, illustrated in Figure 3.

When $t$ is between 0 and 1, our objective is to find an appropriate set of values for $c$ and $\alpha$ that preserve the linear relationship between $t$ and $\mathrm{d}(x_t, x_1)$. This linearity provides a finer control of perturbations over the probability path, as described next. Additional experiments and discussions on the impact of lineared metric path on model convergence and performance are provided in Appendix G.

**Resolution-dependent timestep shifting.** Intuitively, since higher resolutions contain more pixels, more perturbation is needed to alter the signal. To address this, we introduce a time shift parameter $\lambda$, which adjusts the timestep based on the resolution. For any given $t$, we define the shifted timestep $\tilde{t}$

$$\tilde{t} = \frac{t}{t + \lambda(1-t)}, \tag{5}$$

Because our proposed linearized metric path enforces a linear relationship between $t$ and $\mathrm{d}(x_t, x_1)$, we modulate this path using $\lambda$ to accommodate varying data resolutions. For higher resolutions, we set $\lambda > 1$ to create a convex relationship between $\tilde{t}$ and $\mathrm{d}(x_t, x_1)$ that introduces stronger perturbations. For lower resolutions, we set $\lambda < 1$, yielding a concave relationship with more gradual perturbations.

### 3.2.2 Asynchronous Timestep Scheduling

Videos exhibit complex spatiotemporal dynamics and diverse downstream tasks, making task-specific modeling inefficient and resource-intensive. Motivated by diffusion forcing (Chen et al., 2024a), we propose a asynchronous timestep scheduling strategy for multi-task learning. Instead of applying the sample noise level across all frames in a video sequence, we assign perturbation levels **independently** on a per-frame basis. Formally, given a video sequence $\mathbf{F} = \{f^1, f^2, \ldots, f^n\}$ with $n$ frames, we sample a perturbation schedule $\mathbf{T} = \{t^1, t^2, \ldots, t^n\}$, where $t^i \sim \mathcal{U}(0, 1)$. The noisy sequence is then constructed as $\mathbf{F} = \{f^1_{t^1}, f^2_{t^2}, \ldots, f^n_{t^n}\}$, with the diffusion process in Eq. 3 applied frame-wise according to the corresponding $t^i$. This strategy enables fine-grained temporal control over the noise schedule and decouples perturbation levels across frames. As a result, the model can adaptively balance local frame reconstruction with global temporal coherence, facilitating diverse generation objectives, such as text-to-video, image-to-video, video extrapolation and start-end frame control within a unified architecture. More visualizations are provided in Appendix E & F.

### 3.2.3 Training and Sampling

**Training.** We first encode video clips into discrete token sequences using a pretrained tokenizer, yielding $x_1 = (x_1^1, x_1^2 \ldots x_1^i \ldots x_1^F)$, where $F$ denotes the number of video frames and $x_1^i$ denotes the $i$-th frame tokens. At each training step, we uniformly sample timesteps $t^i \in [0, 1]$ independently for each frame $x_1^i$ in the video sequence and obtain a perturbed sequence $x_t \sim p_{t|1}(\cdot \mid x_1)$ via the proposed metric probability path. The model then takes as input the concatenation of text tokens $e$ and $x_t$, and predicts the original visual token sequence $x_1$ by outputting logits over the token vocabulary at each position. The training objective is formulated as the expected cross-entropy between the ground-truth visual tokens and the model's predicted distribution:

$$\mathcal{L} = \mathbb{E}_{t \sim \mathcal{U}[0,1],\, x_1, x_t}\left[-\log p_{1|t}\left(x_1 \mid x_t, e\right)\right]. \tag{6}$$

**Sampling.** During sampling, we follow Gat et al. (2024); Shaul et al. (2025) and employ an Euler solver for efficient and high-quality generation. Specifically, we first uniformly sample $x_0$ from

the full vision vocabulary and feed it into the model to obtain the prediction $\hat{x}_1$. Using Eq. 3 and Eq. 2, we compute the velocity field $u_t(\cdot \mid x_t, \hat{x}_1)$. The estimate of $x_1$ is then iteratively refined as $x_1 \leftarrow x_1 + dt \cdot u_t$, where each step updates the sample along the estimated denoising trajectory. After $T$ such steps, the process yields a clean image or video sequence. Further details on training and sampling are provided in Appendix C.

# 4 Experiment

## 4.1 Experiment Setup

**Datasets.** We leverage a curated selection of high-quality datasets to effectively train UDM models. For text-to-image training, we collect 16M image-text pairs sourced from Unsplash (Unsplash, 2020), DataComp (Gadre et al., 2024), COYO (Byeon et al., 2022), and JourneyDB (Sun et al., 2023). These pairs are filtered by image resolution and aesthetic score, and further supplemented with 14M AI-generated image samples using the FLUX.1 model (Batifol et al., 2025). For text-to-video training, we select 12M video-text pairs from the highest scoring subset of Koala-36M (Wang et al., 2025c) and complement them with 12M internal video-text pairs. The internal videos are captioned using the Emu2-17B model (Sun et al., 2024b) in conjunction with the captioning engine (Diao et al., 2024). We uniformly sample short and long captions during training, with a maximum length of 320 tokens.

**Architectures.** We initialize our visual generation model with weights from a pre-trained LLM. Specifically, we adopt the Qwen3 LLM architecture (Yang et al., 2025a), which natively incorporates QK-Norm (Dehghani et al., 2023) layer to stabilize the multimodal training. To better capture the spatiotemporal structure inherent in videos, we introduce an enhanced M-RoPE (Wang et al., 2024a) that allocates interleaved frequency components across temporal, height, and width dimensions, following the approach of Mogao (Liao et al., 2025). Crucially, unlike Liao et al. (2025), our 3D-RoPE assigns identical positions for texts, ensuring equivalence with the 1D-RoPE (Su et al., 2024). We use the Cosmos (Agarwal et al., 2025) tokenizer to extract image and video tokens, achieving $4\times$ temporal and $8\times8$ spatial compression through a 64K FSQ (Mentzer et al., 2024) codebook. Furthermore, we train an IBQ (Shi et al., 2025) tokenizer for high-resolution image generation, facilitating efficient $16\times16$ spatial compression via a 256-dimensional codebook with 131K entries.

**Diffusion schedulers.** We adopt the Kinetic Optimal Scheduler (Shaul et al., 2025), equipped with a metric-induced probability path specifically designed for the embedding space of vision tokenizers. Following Shaul et al. (2025), we perform a grid search over the path hyperparameters $\alpha$ and $c$, visually inspecting the reconstructed samples for each $(\alpha, c)$ that fully exploit the time interval $[0, 1]$. Eventually, we select $(\alpha, c)$ to $(1.0, 5)$ for the Cosmos tokenizer and $(0.5, 6)$ for our IBQ tokenizer. For conventional uniform diffusion, we use the mixture probability path proposed by Gat et al. (2024). In contrast, for masked diffusion, we adopt the MaskGIT (Chang et al., 2022) scheduler, which has been empirically shown to achieve state-of-the-art performance in both image and video generation models (Kondratyuk et al., 2024; Bai et al., 2025). Following established practice in continuous diffusion models, we default to 25 inference steps for image generation and 50 for video generation.

More implementation details, including Training Details and Evaluation, are provided in Appendix C.

## 4.2 Main results

**UDM rivals Sora-like text-to-video generation models despite using a discrete tokenizer.** Current discrete video tokenizers offer limited spatiotemporal compression and reconstruction quality, posing significant challenges to bidirectional diffusion transformers. However, UDM excels in generating video clips from text, achieving strong performance on the VBench, as shown in Table 1. Compared to Sora-like diffusion models: Vchitect (Fan et al., 2025), Pyramid Flow (Jin et al., 2025), LuminaVideo (Liu et al., 2025a), OpenSora (Peng et al., 2025) and OpenSoraPlan (Lin et al., 2024), UDM matches or exceeds their performance, particularly in the semantic field. These results further underscore the need for a tokenizer that satisfies the imaging quality of state-of-the-art continuous models (Kong et al., 2024; Teng et al., 2025; Ma et al., 2025; Yang et al., 2025b; Wang et al., 2025a).

**UDM emerges frame-conditioned video generation by accurately modeling the future motion.** Prior methods typically adapt text-to-image (Ren et al., 2024; Chen et al., 2024b; Xing et al., 2024) or text-to-video models with a clean first frame for image-to-video generation. In contrast, UDM

Table 1: **Text-to-video evaluation on VBench.** For clarity and to better highlight distinctions between models, we report only the most relevant metrics across the quality and semantic dimensions.

| Model | #params | #videos | Total Score | Quality Score | Semantic Score | Dynamic Degree | Aesthetic Quality | Imaging Quality | Object Class | Multiple Objects | Spatial Relationship | Color | Scene |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| *Continuous models* | | | | | | | | | | | | | |
| LaVie | 3B | 25M | 77.1 | 78.8 | 70.3 | 49.7 | 54.9 | 61.9 | 91.8 | 33.3 | 34.1 | 86.4 | 52.7 |
| Show-1 | 4B | 10M | 78.9 | 80.4 | 73.0 | 44.4 | 57.4 | 58.7 | 93.1 | 45.5 | 53.5 | 86.4 | 47.0 |
| VideoCrafter2 | 2B | 10M | 80.4 | 82.2 | 73.4 | 42.5 | 63.1 | 67.2 | 92.6 | 40.7 | 35.9 | 92.9 | 55.3 |
| Latte-1 | 1B | 25M | 77.3 | 79.2 | 67.6 | 68.9 | 61.6 | 61.9 | 86.5 | 34.5 | 41.5 | 85.3 | 36.3 |
| NOVA | 0.6B | 20M | 80.1 | 80.4 | 79.1 | - | 59.4 | - | 92.0 | 77.5 | 77.5 | - | 54.1 |
| Vchitect-2.0 | 2B | 134M | 81.6 | 82.5 | 77.8 | 58.3 | 61.5 | 65.6 | 87.8 | 69.4 | 54.6 | 86.9 | 57.5 |
| Pyramid Flow | 2B | 10M | 81.7 | 84.7 | 69.6 | 64.6 | 63.3 | 65.0 | 86.7 | 50.7 | 59.5 | 82.9 | 43.2 |
| LuminaVideo | 2B | 12M | 83.0 | 83.9 | 79.3 | 67.1 | 62.3 | 64.6 | 91.0 | 68.3 | 67.3 | 90.2 | 56.1 |
| OpenSoraPlan v1.5 | 8B | 40M | 83.0 | 84.2 | 78.2 | 64.4 | 66.9 | 68.5 | 91.9 | 70.7 | 80.1 | 81.8 | 52.1 |
| OpenSora 2.0 | 11B | 85M | 83.6 | 84.4 | 80.3 | 56.4 | 65.3 | 65.7 | 94.6 | 78.0 | 76.8 | 86.3 | 53.4 |
| MAGI-1 | 24B | - | 81.8 | 84.7 | 70.4 | 72.5 | 59.3 | 65.3 | 84.1 | 50.6 | 73.0 | 87.5 | 28.9 |
| Step-Video | 30B | - | 81.8 | 84.5 | 71.3 | 53.1 | 61.2 | 70.6 | 80.6 | 50.6 | 71.5 | 88.3 | 24.4 |
| CogVideoX1.5 | 5B | - | 82.0 | 82.7 | 79.2 | 56.2 | 62.1 | 65.3 | 83.4 | 65.3 | 79.4 | 88.4 | 53.3 |
| HunyuanVideo | 13B | - | 83.2 | 85.1 | 75.8 | 70.8 | 60.4 | 67.6 | 86.1 | 68.6 | 68.7 | 91.6 | 53.9 |
| Wan2.1 | 14B | - | 83.7 | 85.6 | 76.1 | 65.5 | 66.1 | 69.4 | 86.3 | 69.6 | 75.4 | 88.6 | 45.8 |
| *Discrete models* | | | | | | | | | | | | | |
| Lumos-1 | 3.6B | 10M | 78.3 | 79.5 | 73.5 | - | - | 58.0 | 90.1 | - | - | 82.0 | - |
| Emu3 | 8B | - | 81.0 | 84.1 | 68.4 | 79.3 | 59.6 | 62.6 | 86.2 | 44.6 | 68.7 | 88.3 | 37.1 |
| UDM | 1.7B | 24M | 82.4 | 83.4 | 78.5 | 81.4 | 63.1 | 62.2 | 93.4 | 70.6 | 62.1 | 90.7 | 52.3 |

seamlessly integrates asynchronous frame conditions, enabling zero-shot generalization for this task. As depicted in Table 2, UDM excels in camera control and subject movement versus specialized frame-conditioned models (Agarwal et al., 2025; Yu et al., 2025; Wang et al., 2025a; Liu et al., 2025b). Our results demonstrate that diffusion forcing effectively generalizes to image-to-video generation, pushing the boundaries of autoregressive discrete video generation models without causal attention.

Table 2: **Image-to-video evaluation on VBench++.** To evaluate temporal consistency, we focus on image-to-video (I2V) metrics of visual similarity between each generated frame and reference image.

| Model | #params | #videos | Total Score | Quality Score | I2V Score | Dynamic Degree | Aesthetic Quality | Imaging Quality | Camera Motion | I2V Subject Consistency | I2V Background Consistency |
|---|---|---|---|---|---|---|---|---|---|---|---|
| *Continuous models* | | | | | | | | | | | |
| ConsistI2V | 2B | 10M | 84.1 | 76.2 | 91.9 | 18.6 | 59.0 | 66.9 | 33.9 | 95.8 | 96.0 |
| I2VGen-XL | 2B | 35M | 85.3 | 78.4 | 92.1 | 26.1 | 64.8 | 69.1 | 18.5 | 96.5 | 96.8 |
| SEINE | 3B | 25M | 85.5 | 78.4 | 92.7 | 27.1 | 64.6 | 71.4 | 21.0 | 97.2 | 97.0 |
| DynamiCrafter | 2B | 10M | 86.9 | 80.5 | 93.5 | 69.7 | 60.9 | 68.6 | 31.2 | 97.2 | 97.4 |
| Cosmos | 13B | 100M | 84.2 | 75.8 | 92.6 | 18.7 | 55.8 | 59.9 | 25.4 | 96.0 | 97.4 |
| VideoMAR | 1.4B | 0.5M | 84.8 | 75.6 | 94.0 | 11.0 | 55.8 | 62.3 | 21.6 | 97.9 | 98.4 |
| CogVideoX | 5B | - | 86.7 | 78.6 | 94.8 | 33.2 | 61.9 | 70.0 | 67.7 | 97.2 | 96.7 |
| HunyuanVideo | 13B | - | 86.8 | 78.5 | 95.1 | 22.2 | 62.6 | 70.1 | 49.9 | 98.5 | 97.4 |
| Wan2.1 | 14B | - | 86.9 | 80.8 | 92.9 | 51.4 | 64.8 | 70.4 | 34.8 | 97.0 | 96.4 |
| Pusa | 14B | - | 87.3 | 79.8 | 94.8 | 52.6 | 63.2 | 68.3 | 29.5 | 97.6 | 99.2 |
| Step-Video | 30B | - | 88.4 | 81.2 | 95.5 | 48.8 | 62.3 | 70.4 | 49.2 | 97.9 | 98.5 |
| MAGI-1 | 24B | - | 89.3 | 82.4 | 96.1 | 68.2 | 64.7 | 69.7 | 50.9 | 98.4 | 99.0 |
| *Discrete models* | | | | | | | | | | | |
| Lumos-1 | 3.6B | 10M | 84.7 | 76.1 | 93.3 | - | - | 69.2 | - | 97.4 | 97.4 |
| UDM | 1.7B | 24M | 86.2 | 79.8 | 92.6 | 65.3 | 57.4 | 64.2 | 37.6 | 96.1 | 96.5 |

**UDM performs on par with the state-of-the-art models in generating high-resolution images.** We compare UDM against continuous models in Table 3, encompassing specialist architectures: SDXL (Podell et al., 2024), SD3 (Esser et al., 2024), FLUX (Batifol et al., 2025), SANA (Xie et al., 2025a) and NOVA (Deng et al., 2025b), as well as unified architectures: Mogao (Liao et al., 2025), Bagel (Deng et al., 2025a), OmniGen2 (Wu et al., 2025b) and Show-o2 (Xie et al., 2025d). Through joint modeling of discrete text and visual tokens, UDM demonstrates strong text-image alignment. For example, on the DPG-Bench, UDM reaches a leading overall score with dense text prompts. This strong performance is consistently sustained on the GenEval when using the rewritten prompts. At high resolutions, UDM surpasses the autoregressive (Wang et al., 2024b; Han et al., 2025; Chen et al., 2025b) and masked diffusion (Bai et al., 2025; Yuan et al., 2025) approaches in efficiency, effectively reducing inference steps through iterative refinement while preserving fine-grained detail.

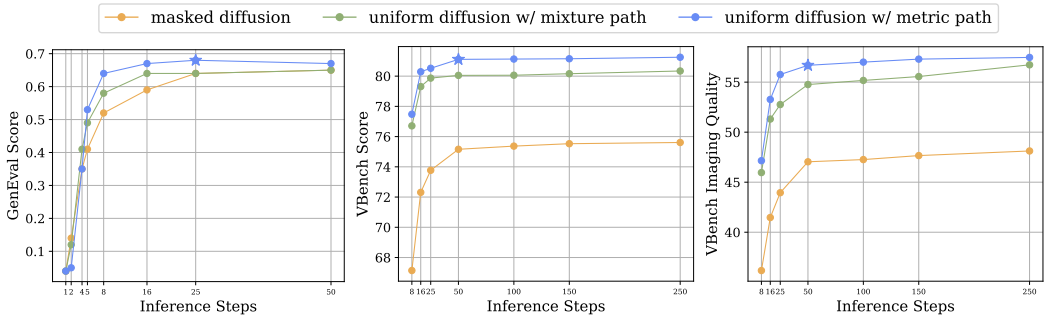
### 4.3 Ablation Study

**Effectiveness of iterative refinement for visual generation.** Discrete diffusion models inherently incur elevated sampling errors, as exhibited in prior studies (Tang et al., 2022; Feng et al., 2025).

Table 3: **Text-to-image evaluation on DPG-Bench and GenEval.** We prefer the DPG-Bench metrics to mitigate potential prompt template leakage concerns (Xie et al., 2025b) associated with GenEval. † refers to the methods using rewritten GenEval prompts for clearer position and attribute guidance.

| Model | ModelSpec | | DPG-Bench | | | | GenEval | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | #params | #images | Overall | Entity | Attribute | Relation | Overall | Single | Two | Counting | Colors | Position | ColorAttr |
| *Continuous models* | | | | | | | | | | | | | |
| SDXL | 2.6B | - | 74.7 | 82.4 | 80.9 | 86.8 | 0.55 | 0.98 | 0.44 | 0.39 | 0.85 | 0.15 | 0.23 |
| SD3 | 2B | - | 84.1 | 91.0 | 88.8 | 80.7 | 0.62 | 0.98 | 0.74 | 0.63 | 0.67 | 0.34 | 0.36 |
| FLUX.1-dev | 12B | - | 84.9 | - | - | - | 0.68 | 0.99 | 0.85 | 0.74 | 0.79 | 0.21 | 0.48 |
| NOVA | 1.4B | 600M | 83.0 | 88.7 | 86.4 | 91.9 | 0.71 | 0.99 | 0.91 | 0.62 | 0.85 | 0.33 | 0.56 |
| SANA-1.5† | 4.8B | 50M | 84.7 | - | - | - | 0.81 | 0.99 | 0.93 | 0.86 | 0.84 | 0.59 | 0.65 |
| OmniGen2 | 7B | - | 83.6 | 88.8 | 90.2 | 89.4 | 0.80 | 1.00 | 0.95 | 0.64 | 0.88 | 0.55 | 0.76 |
| Mogao† | 7B | - | 84.3 | 90.0 | 88.3 | 93.2 | 0.89 | 1.00 | 0.97 | 0.83 | 0.93 | 0.84 | 0.80 |
| Bagel | 14B | - | 85.1 | 90.4 | 91.3 | 90.8 | 0.82 | 0.99 | 0.94 | 0.81 | 0.88 | 0.64 | 0.63 |
| Show-o2† | 7B | 66M | 86.1 | 91.8 | 90.0 | 91.8 | 0.76 | 1.00 | 0.87 | 0.58 | 0.92 | 0.52 | 0.62 |
| *Discrete models* | | | | | | | | | | | | | |
| Show-o | 1.3B | 2B | 67.3 | 75.4 | 78.0 | 84.5 | 0.68 | 0.98 | 0.80 | 0.66 | 0.84 | 0.31 | 0.50 |
| Emu3† | 8B | - | 81.6 | 87.2 | 86.3 | 90.6 | 0.66 | 0.99 | 0.81 | 0.42 | 0.80 | 0.49 | 0.45 |
| FUDOKI | 1.5B | 13M | 83.6 | 89.7 | 88.1 | 93.7 | 0.77 | 0.96 | 0.85 | 0.56 | 0.88 | 0.68 | 0.67 |
| Janus-Pro | 7B | 72M | 84.2 | 88.9 | 89.4 | 89.3 | 0.80 | 0.99 | 0.89 | 0.59 | 0.90 | 0.79 | 0.66 |
| Meissonic | 1B | 210M | - | - | - | - | 0.54 | 0.99 | 0.66 | 0.42 | 0.86 | 0.10 | 0.22 |
| Lumos-1† | 3.6B | 60M | - | - | - | - | 0.66 | 0.95 | 0.80 | 0.46 | 0.81 | 0.48 | 0.48 |
| Infinity† | 2B | - | 83.5 | - | - | 90.8 | 0.73 | 0.99 | 0.85 | 0.64 | 0.84 | 0.49 | 0.57 |
| UDM (512×320) | 1.7B | 30M | 82.5 | 88.3 | 86.4 | 92.9 | 0.64 | 0.99 | 0.83 | 0.47 | 0.83 | 0.30 | 0.41 |
| UDM (1024×1024) | 1.7B | 30M | 86.0 | 91.5 | 89.6 | 94.7 | 0.68 | 0.99 | 0.92 | 0.63 | 0.86 | 0.25 | 0.40 |
| UDM† (1024×1024) | 1.7B | 30M | - | - | - | - | 0.80 | 1.00 | 0.92 | 0.64 | 0.89 | 0.67 | 0.69 |

To systematically investigate this issue in image and video generation, we train three variants of the discrete diffusion model, assessing performance across insufficient and excessive sampling regimes. Figure 4 compares key performance metrics of text-to-image models on GenEval and text-to-video models on VBench, with all models evaluated after being trained for an identical number of iterations. In the image generation task, which is characterized by low structural redundancy, all three models can generate feasible images within the conventional 25 inference steps. Without iterative refinement, reducing the number of steps substantially decreases the GenEval score in masked diffusion sampling. As we progress into video generation, a task rich in contextual redundancy, it becomes essential to correct sampling errors at each step, ensuring temporal coherence and visual fidelity across frames.




(a) Text-to-image performance on GenEval. (b) Text-to-video performance on VBench.

Figure 4: **Sampling performance across inference steps.** Using the Cosmos tokenizer, we evaluate the image samples at 256×256 (∼1K tokens) and the video samples at 25×384×240 (∼10K tokens).

**Effectiveness of timestep conditioning for uniform diffusion.** Recent work explores time-agnostic (i.e., noise-unconditional) methods for both continuous diffusion (Sun et al., 2025; Tang et al., 2025) and masked diffusion (Zheng et al., 2025; Ou et al., 2025), effectively narrowing the architectural gap between diffusion transformers (DiTs) and LLMs. In this context, we analyze whether timestep conditioning remains indispensable for uniform diffusion. The results are illustrated in Figure 5. Specifically, we train three model variants with distinct conditioning strategies and evaluate GenEval across training iterations. After one epoch (∼30K iterations), embedding or prompting with the timestep provides no measurable benefit. Notably, timestep embedding can degrade performance as their variance increases, potentially disrupting token embedding and compromising training stability.

**Effectiveness of timestep shifting for video generation.** As outlined in Section 4.1, our probability path is designed to maximize the time interval. In line with continuous models (Kong et al., 2024;

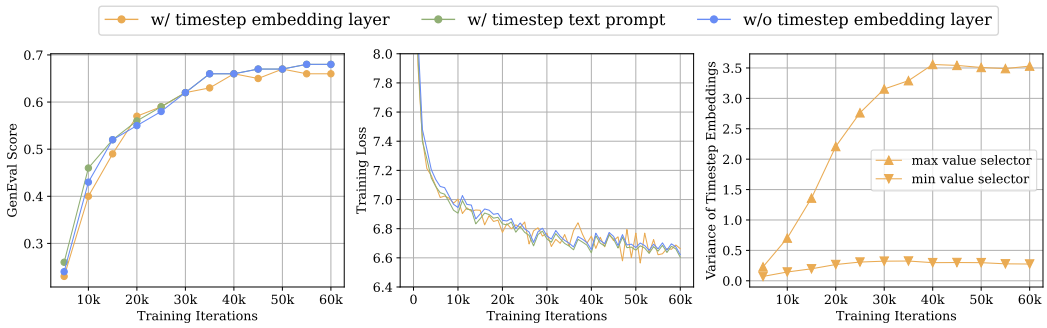



Figure 5: **Model metrics across training iterations.** We sample 256×256 images for evaluation.

Wang et al., 2025a; Liu et al., 2025a), the optimal SNR schedule should be tailored with video size. To study the impact of the SNR schedule on video generation, we train four text-to-video models with divergent timestep shifting and evaluate their performance using the respective value on VBench. Figure 6 presents our shifting schedules, accompanied by their evaluation metrics and visualizations. Surprisingly, the shifting strategy proposed by Esser et al. (2024) demonstrates strong effectiveness for uniform diffusion, empowering UDM to match the performance of its continuous counterparts.

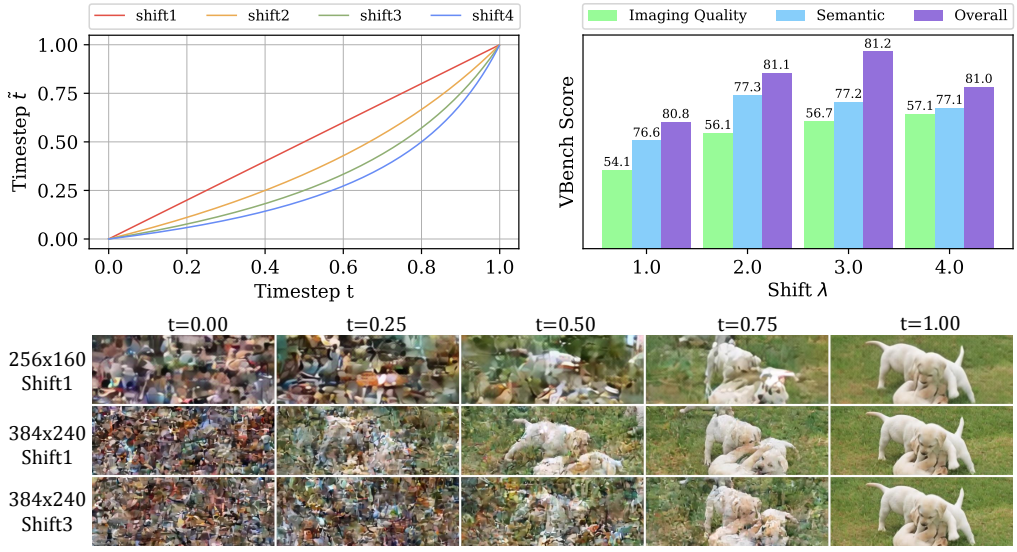



Figure 6: **Timestep shifting across SNR schedules.** We sample 25×384×240 videos for evaluation.

## 5 Conclusion

In this work, we revisited discrete generative modeling for video synthesis and introduced **UDM**, a uniform discrete diffusion framework with a metric path that bridges discrete and continuous paradigms. UDM employs two key innovations, Linearized Metric-Path and Resolution-dependent Timestep Shifting, to provide fine-grained control over perturbations, enabling stable and scalable training for both high-resolution and long-duration video generation. On top of this, our asynchronous temporal scheduling strategy unifies multiple tasks, such as video interpolation and image-to-video synthesis, within a single model. Extensive experiments show that UDM not only consistently surpasses prior discrete approaches but also achieves results comparable to state-of-the-art continuous diffusion models. We believe this work marks a step toward unifying discrete and continuous paradigms and offers a promising direction for scalable, versatile, and efficient visual generation.

## ETHICS AND REPRODUCIBILITY STATEMENT

This work aims to advance discrete generative modeling for video generation through Uniform Discrete diffusion with Metric path. No personal, private, or sensitive information is included in the datasets or experiments, and no ethical risks are associated with this study. With respect to reproducibility, we affirm our commitment to ensuring that all reported results can be faithfully reproduced, and we will provide the necessary resources and documentation to facilitate replication.

## REFERENCES


Niket Agarwal, Arslan Ali, Maciej Bala, Yogesh Balaji, Erik Barker, Tiffany Cai, Prithvijit Chattopadhyay, Yongxin Chen, Yin Cui, Yifan Ding, et al. Cosmos world foundation model platform for physical ai. *arXiv preprint arXiv:2501.03575*, 2025.

Jinbin Bai, Tian Ye, Wei Chow, Enxin Song, Xiangtai Li, Zhen Dong, Lei Zhu, and Shuicheng Yan. Meissonic: Revitalizing masked generative transformers for efficient high-resolution text-to-image synthesis. In *ICLR*, 2025.

Jason Baldridge, Jakob Bauer, Mukul Bhutani, Nicole Brichtova, Andrew Bunner, Kelvin Chan, Yichang Chen, Sander Dieleman, Yuqing Du, Zach Eaton-Rosen, et al. Imagen 3. *arXiv preprint arXiv:2408.07009*, 2024.

Stephen Batifol, Andreas Blattmann, Frederic Boesel, Saksham Consul, Cyril Diagne, Tim Dockhorn, Jack English, Zion English, Patrick Esser, et al. Flux. 1 kontext: Flow matching for in-context image generation and editing in latent space. *arXiv preprint arXiv:2506.15742*, 2025.

James Betker, Gabriel Goh, Li Jing, Tim Brooks, Jianfeng Wang, Linjie Li, Long Ouyang, Juntang Zhuang, Joyce Lee, Yufei Guo, et al. Improving image generation with better captions, 2023.

Andrew Brock, Jeff Donahue, and Karen Simonyan. Large scale gan training for high fidelity natural image synthesis. *arXiv preprint arXiv:1809.11096*, 2018.

Tim Brooks, Bill Peebles, Connor Holmes, Will DePue, Yufei Guo, Li Jing, David Schnurr, Joe Taylor, Troy Luhman, Eric Luhman, et al. Video generation models as world simulators, 2024.

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

Minwoo Byeon, Beomhee Park, Haecheon Kim, Sungjun Lee, Woonhyuk Baek, and Saehoon Kim. Coyo-700m: Image-text pair dataset, 2022.

Huiwen Chang, Han Zhang, Lu Jiang, Ce Liu, and William T Freeman. Maskgit: Masked generative image transformer. In *CVPR*, 2022.

Huiwen Chang, Han Zhang, Jarred Barber, AJ Maschinot, Jose Lezama, Lu Jiang, Ming-Hsuan Yang, Kevin Murphy, William T Freeman, Michael Rubinstein, et al. Muse: Text-to-image generation via masked generative transformers. In *ICML*, 2023.

Boyuan Chen, Diego Martí Monsó, Yilun Du, Max Simchowitz, Russ Tedrake, and Vincent Sitzmann. Diffusion forcing: Next-token prediction meets full-sequence diffusion. *Advances in Neural Information Processing Systems*, 2024a.

Guibin Chen, Dixuan Lin, Jiangping Yang, Chunze Lin, Junchen Zhu, Mingyuan Fan, Hao Zhang, Sheng Chen, Zheng Chen, Chengcheng Ma, et al. Skyreels-v2: Infinite-length film generative model. *arXiv preprint arXiv:2504.13074*, 2025a.

Xiaokang Chen, Zhiyu Wu, Xingchao Liu, Zizheng Pan, Wen Liu, Zhenda Xie, Xingkai Yu, and Chong Ruan. Janus-pro: Unified multimodal understanding and generation with data and model scaling. *arXiv preprint arXiv:2501.17811*, 2025b.

Xinyuan Chen, Yaohui Wang, Lingjun Zhang, Shaobin Zhuang, Xin Ma, Jiashuo Yu, Yali Wang, Dahua Lin, Yu Qiao, and Ziwei Liu. Seine: Short-to-long video diffusion model for generative transition and prediction. In *ICLR*, 2024b.

Mostafa Dehghani, Josip Djolonga, Basil Mustafa, Piotr Padlewski, Jonathan Heek, Justin Gilmer, Andreas Peter Steiner, Mathilde Caron, Robert Geirhos, Ibrahim Alabdulmohsin, et al. Scaling vision transformers to 22 billion parameters. In *ICML*, 2023.

Chaorui Deng, Deyao Zhu, Kunchang Li, Chenhui Gou, Feng Li, Zeyu Wang, Shu Zhong, Weihao Yu, Xiaonan Nie, Ziang Song, et al. Emerging properties in unified multimodal pretraining. *arXiv preprint arXiv:2505.14683*, 2025a.

Haoge Deng, Ting Pan, Haiwen Diao, Zhengxiong Luo, Yufeng Cui, Huchuan Lu, Shiguang Shan, Yonggang Qi, and Xinlong Wang. Autoregressive video generation without vector quantization. In *ICLR*, 2025b.

Haiwen Diao, Yufeng Cui, Xiaotong Li, Yueze Wang, Huchuan Lu, and Xinlong Wang. Unveiling encoder-free vision-language models. In *NeurIPS*, 2024.

Ming Ding, Zhuoyi Yang, Wenyi Hong, Wendi Zheng, Chang Zhou, Da Yin, Junyang Lin, Xu Zou, Zhou Shao, Hongxia Yang, et al. Cogview: Mastering text-to-image generation via transformers. In *NeurIPS*, 2021.

Laurent Dinh, David Krueger, and Yoshua Bengio. Nice: Non-linear independent components estimation. *arXiv preprint arXiv:1410.8516*, 2014.

Laurent Dinh, Jascha Sohl-Dickstein, and Samy Bengio. Density estimation using real nvp. *arXiv preprint arXiv:1605.08803*, 2016.

Patrick Esser, Robin Rombach, and Bjorn Ommer. Taming transformers for high-resolution image synthesis. In *CVPR*, 2021.

Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, et al. Scaling rectified flow transformers for high-resolution image synthesis. In *ICML*, 2024.

Weichen Fan, Chenyang Si, Junhao Song, Zhenyu Yang, Yinan He, Long Zhuo, Ziqi Huang, Ziyue Dong, Jingwen He, Dongwei Pan, et al. Vchitect-2.0: Parallel transformer for scaling up video diffusion models. *arXiv preprint arXiv:2501.08453*, 2025.

Guhao Feng, Yihan Geng, Jian Guan, Wei Wu, Liwei Wang, and Di He. Theoretical benefit and limitation of diffusion language model. In *NeurIPS*, 2025.

Samir Yitzhak Gadre, Gabriel Ilharco, Alex Fang, Jonathan Hayase, Georgios Smyrnis, Thao Nguyen, Ryan Marten, Mitchell Wortsman, Dhruba Ghosh, Jieyu Zhang, et al. Datacomp: In search of the next generation of multimodal datasets. In *NeurIPS*, 2024.

Yu Gao, Lixue Gong, Qiushan Guo, Xiaoxia Hou, Zhichao Lai, Fanshi Li, Liang Li, Xiaochen Lian, Chao Liao, Liyang Liu, et al. Seedream 3.0 technical report. *arXiv preprint arXiv:2504.11346*, 2025a.

Yu Gao, Haoyuan Guo, Tuyen Hoang, Weilin Huang, Lu Jiang, Fangyuan Kong, Huixia Li, Jiashi Li, Liang Li, Xiaojie Li, et al. Seedance 1.0: Exploring the boundaries of video generation models. *arXiv preprint arXiv:2506.09113*, 2025b.

Itai Gat, Tal Remez, Neta Shaul, Felix Kreuk, Ricky TQ Chen, Gabriel Synnaeve, Yossi Adi, and Yaron Lipman. Discrete flow matching. In *NeurIPS*, 2024.

Dhruba Ghosh, Hannaneh Hajishirzi, and Ludwig Schmidt. Geneval: An object-focused framework for evaluating text-to-image alignment. In *NeurIPS*, 2024.

Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks. *Communications of the ACM*, 63(11):139–144, 2020.

Jian Han, Jinlai Liu, Yi Jiang, Bin Yan, Yuqi Zhang, Zehuan Yuan, Bingyue Peng, and Xiaobing Liu. Infinity: Scaling bitwise autoregressive modeling for high-resolution image synthesis. In *CVPR*, 2025.

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

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *NeurIPS*, 2020.

Emiel Hoogeboom, Jonathan Heek, and Tim Salimans. simple diffusion: End-to-end diffusion for high resolution images. In *ICML*, 2023.

Xiwei Hu, Rui Wang, Yixiao Fang, Bin Fu, Pei Cheng, and Gang Yu. Ella: Equip diffusion models with llm for enhanced semantic alignment. *arXiv preprint arXiv:2403.05135*, 2024.

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

Ziqi Huang, Fan Zhang, Xiaojie Xu, Yinan He, Jiashuo Yu, Ziyue Dong, Qianli Ma, Nattapol Chanpaisit, Chenyang Si, Yuming Jiang, et al. Vbench++: Comprehensive and versatile benchmark suite for video generative models. *arXiv preprint arXiv:2411.13503*, 2024b.

Yang Jin, Zhicheng Sun, Ningyuan Li, Kun Xu, Kun Xu, Hao Jiang, Nan Zhuang, Quzhe Huang, Yang Song, Yadong MU, and Zhouchen Lin. Pyramidal flow matching for efficient video generative modeling. In *ICLR*, 2025.

Nal Kalchbrenner, Aäron Oord, Karen Simonyan, Ivo Danihelka, Oriol Vinyals, Alex Graves, and Koray Kavukcuoglu. Video pixel networks. In *ICML*, 2017.

Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Analyzing and improving the image quality of stylegan. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 8110–8119, 2020.

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

Dan Kondratyuk, Lijun Yu, Xiuye Gu, Jose Lezama, Jonathan Huang, Grant Schindler, Rachel Hornung, Vighnesh Birodkar, Jimmy Yan, Ming-Chang Chiu, et al. Videopoet: A large language model for zero-shot video generation. In *ICML*, 2024.

Weijie Kong, Qi Tian, Zijian Zhang, Rox Min, Zuozhuo Dai, Jin Zhou, Jiangfeng Xiong, Xin Li, Bo Wu, Jianwei Zhang, et al. Hunyuanvideo: A systematic framework for large video generative models. *arXiv preprint arXiv:2412.03603*, 2024.

Kuaishou. Kling ai, 2024. URL https://klingai.com/.

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

Chao Liao, Liyang Liu, Xun Wang, Zhengxiong Luo, Xinyu Zhang, Wenliang Zhao, Jie Wu, Liang Li, Zhi Tian, and Weilin Huang. Mogao: An omni foundation model for interleaved multi-modal generation. *arXiv preprint arXiv:2505.05472*, 2025.

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

Dongyang Liu, Shicheng Li, Yutong Liu, Zhen Li, Kai Wang, Xinyue Li, Qi Qin, Yufei Liu, Yi Xin, Zhongyu Li, et al. Lumina-video: Efficient and flexible video generation with multi-scale next-dit. *arXiv preprint arXiv:2502.06782*, 2025a.

Yaofang Liu, Yumeng Ren, Xiaodong Cun, Aitor Artola, Yang Liu, Tieyong Zeng, Raymond H Chan, and Jean-michel Morel. Redefining temporal modeling in video diffusion: The vectorized timestep approach. *arXiv preprint arXiv:2410.03160*, 2024.

Yaofang Liu, Yumeng Ren, Aitor Artola, Yuxuan Hu, Xiaodong Cun, Xiaotong Zhao, Alan Zhao, Raymond H Chan, Suiyun Zhang, Rui Liu, et al. Pusa v1.0: Surpassing wan-i2v with $500 training cost by vectorized timestep adaptation. *arXiv preprint arXiv:2507.16116*, 2025b.

Ilya Loshchilov and Frank Hutter. Sgdr: Stochastic gradient descent with warm restarts. In *ICLR*, 2017.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *ICLR*, 2019.

Guoqing Ma, Haoyang Huang, Kun Yan, Liangyu Chen, Nan Duan, Shengming Yin, Changyi Wan, Ranchen Ming, Xiaoniu Song, Xing Chen, et al. Step-video-t2v technical report: The practice, challenges, and future of video foundation model. *arXiv preprint arXiv:2502.10248*, 2025.

Fabian Mentzer, David Minnen, Eirikur Agustsson, and Michael Tschannen. Finite scalar quantization: VQ-VAE made simple. In *ICLR*, 2024.

Aaron van den Oord, Oriol Vinyals, and Koray Kavukcuoglu. Neural discrete representation learning. In *NeurIPS*, 2017.

Jingyang Ou, Shen Nie, Kaiwen Xue, Fengqi Zhu, Jiacheng Sun, Zhenguo Li, and Chongxuan Li. Your absorbing discrete diffusion secretly models the conditional distributions of clean data. In *ICLR*, 2025.

Xiangyu Peng, Zangwei Zheng, Chenhui Shen, Tom Young, Xinying Guo, Binluo Wang, Hang Xu, Hongxin Liu, Mingyan Jiang, Wenjun Li, et al. Opensora 2.0: Training a commercial-level video generation model in $200k. *arXiv preprint arXiv:2503.09642*, 2025.

Dustin Podell, Zion English, Kyle Lacey, Andreas Blattmann, Tim Dockhorn, Jonas Müller, Joe Penna, and Robin Rombach. Sdxl: Improving latent diffusion models for high-resolution image synthesis. In *ICLR*, 2024.

Alec Radford, Karthik Narasimhan, Tim Salimans, Ilya Sutskever, et al. Improving language understanding by generative pre-training. *OpenAI Blog*, 2018.

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

Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. Zero-shot text-to-image generation. In *ICML*, 2021.

Scott Reed, Aäron Oord, Nal Kalchbrenner, Sergio Gómez Colmenarejo, Ziyu Wang, Yutian Chen, Dan Belov, and Nando Freitas. Parallel multiscale autoregressive density estimation. In *International conference on machine learning*, pp. 2912–2921. PMLR, 2017.

Weiming Ren, Huan Yang, Ge Zhang, Cong Wei, Xinrun Du, Wenhao Huang, and Wenhu Chen. Consisti2v: Enhancing visual consistency for image-to-video generation. *TMLR*, 2024.

Neta Shaul, Itai Gat, Marton Havasi, Daniel Severo, Anuroop Sriram, Peter Holderrieth, Brian Karrer, Yaron Lipman, and Ricky T. Q. Chen. Flow matching with general discrete paths: A kinetic-optimal perspective. In *ICLR*, 2025.

Fengyuan Shi, Zhuoyan Luo, Yixiao Ge, Yujiu Yang, Ying Shan, and Limin Wang. Scalable image tokenization with index backpropagation quantization. In *ICCV*, 2025.

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

Jianlin Su, Murtadha Ahmed, Yu Lu, Shengfeng Pan, Wen Bo, and Yunfeng Liu. Roformer: Enhanced transformer with rotary position embedding. *Neurocomputing*, 568:127063, 2024.

Keqiang Sun, Junting Pan, Yuying Ge, Hao Li, Haodong Duan, Xiaoshi Wu, Renrui Zhang, Aojun Zhou, Zipeng Qin, Yi Wang, Jifeng Dai, Yu Qiao, Limin Wang, and Hongsheng Li. JourneyDB: A benchmark for generative image understanding. In *NeurIPS*, 2023.

Peize Sun, Yi Jiang, Shoufa Chen, Shilong Zhang, Bingyue Peng, Ping Luo, and Zehuan Yuan. Autoregressive model beats diffusion: Llama for scalable image generation. *arXiv preprint arXiv:2406.06525*, 2024a.

Qiao Sun, Zhicheng Jiang, Hanhong Zhao, and Kaiming He. Is noise conditioning necessary for denoising generative models? In *ICML*, 2025.

Quan Sun, Yufeng Cui, Xiaosong Zhang, Fan Zhang, Qiying Yu, Yueze Wang, Yongming Rao, Jingjing Liu, Tiejun Huang, and Xinlong Wang. Generative multimodal models are in-context learners. In *CVPR*, 2024b.

Bingda Tang, Boyang Zheng, Sayak Paul, and Saining Xie. Exploring the deep fusion of large language models and diffusion transformers for text-to-image synthesis. In *CVPR*, 2025.

Zhicong Tang, Shuyang Gu, Jianmin Bao, Dong Chen, and Fang Wen. Improved vector quantized diffusion models. *arXiv preprint arXiv:2205.16007*, 2022.

Hansi Teng, Hongyu Jia, Lei Sun, Lingzhi Li, Maolin Li, Mingqiu Tang, Shuai Han, Tianning Zhang, WQ Zhang, Weifeng Luo, et al. Magi-1: Autoregressive video generation at scale. *arXiv preprint arXiv:2505.13211*, 2025.

Unsplash. Unsplash dataset, 2020.

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

Jin Wang, Yao Lai, Aoxue Li, Shifeng Zhang, Jiacheng Sun, Ning Kang, Chengyue Wu, Zhenguo Li, and Ping Luo. Fudoki: Discrete flow-based unified understanding and generation via kinetic-optimal velocities. In *NeurIPS*, 2025b.

Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, et al. Qwen2-vl: Enhancing vision-language model's perception of the world at any resolution. *arXiv preprint arXiv:2409.12191*, 2024a.

Qiuheng Wang, Yukai Shi, Jiarong Ou, Rui Chen, Ke Lin, Jiahao Wang, Boyuan Jiang, Haotian Yang, Mingwu Zheng, Xin Tao, et al. Koala-36m: A large-scale video dataset improving consistency between fine-grained conditions and video content. In *CVPR*, 2025c.

Xinlong Wang, Xiaosong Zhang, Zhengxiong Luo, Quan Sun, Yufeng Cui, Jinsheng Wang, Fan Zhang, Yueze Wang, Zhen Li, Qiying Yu, et al. Emu3: Next-token prediction is all you need. *arXiv preprint arXiv:2409.18869*, 2024b.

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

Chenfei Wu, Jiahao Li, Jingren Zhou, Junyang Lin, Kaiyuan Gao, Kun Yan, Sheng-ming Yin, Shuai Bai, Xiao Xu, Yilei Chen, et al. Qwen-image technical report. *arXiv preprint arXiv:2508.02324*, 2025a.

Chenyuan Wu, Pengfei Zheng, Ruiran Yan, Shitao Xiao, Xin Luo, Yueze Wang, Wanli Li, Xiyan Jiang, Yexin Liu, Junjie Zhou, et al. Omnigen2: Exploration to advanced multimodal generation. *arXiv preprint arXiv:2506.18871*, 2025b.

Enze Xie, Junsong Chen, Yuyang Zhao, Jincheng YU, Ligeng Zhu, Yujun Lin, Zhekai Zhang, Muyang Li, Junyu Chen, Han Cai, Bingchen Liu, Daquan Zhou, and Song Han. Sana 1.5: Efficient scaling of training-time and inference-time compute in linear diffusion transformer. In *ICML*, 2025a.

Ji Xie, Trevor Darrell, Luke Zettlemoyer, and XuDong Wang. Reconstruction alignment improves unified multimodal models. *arXiv preprint arXiv:2509.07295*, 2025b.

Jinheng Xie, Weijia Mao, Zechen Bai, David Junhao Zhang, Weihao Wang, Kevin Qinghong Lin, Yuchao Gu, Zhijie Chen, Zhenheng Yang, and Mike Zheng Shou. Show-o: One single transformer to unify multimodal understanding and generation. In *ICLR*, 2025c.

Jinheng Xie, Zhenheng Yang, and Mike Zheng Shou. Show-o2: Improved native unified multimodal models. In *NeurIPS*, 2025d.

Jinbo Xing, Menghan Xia, Yong Zhang, Hao Chen, Wangbo Yu, Hanyuan Liu, Gongye Liu, Xintao Wang, Ying Shan, and Tien-Tsin Wong. Dynamicrafter: Animating open-domain images with video diffusion priors. In *ECCV*, 2024.

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

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025a.

Zhuoyi Yang, Jiayan Teng, Wendi Zheng, Ming Ding, Shiyu Huang, Jiazheng Xu, Yuanming Yang, Wenyi Hong, Xiaohan Zhang, Guanyu Feng, Da Yin, Yuxuan.Zhang, Weihan Wang, Yean Cheng, Bin Xu, Xiaotao Gu, Yuxiao Dong, and Jie Tang. Cogvideox: Text-to-video diffusion models with an expert transformer. In *ICLR*, 2025b.

Hu Yu, Biao Gong, Hangjie Yuan, DanDan Zheng, Weilong Chai, Jingdong Chen, Kecheng Zheng, and Feng Zhao. Videomar: Autoregressive video generatio with continuous tokens. In *NeurIPS*, 2025.

Jiahui Yu, Yuanzhong Xu, Jing Yu Koh, Thang Luong, Gunjan Baid, Zirui Wang, Vijay Vasudevan, Alexander Ku, Yinfei Yang, Burcu Karagol Ayan, Ben Hutchinson, Wei Han, Zarana Parekh, Xin Li, Han Zhang, Jason Baldridge, and Yonghui Wu. Scaling autoregressive models for content-rich text-to-image generation. *TMLR*, 2022.

Lijun Yu, Yong Cheng, Kihyuk Sohn, José Lezama, Han Zhang, Huiwen Chang, Alexander G Hauptmann, Ming-Hsuan Yang, Yuan Hao, Irfan Essa, et al. Magvit: Masked generative video transformer. In *CVPR*, 2023.

Hangjie Yuan, Weihua Chen, Jun Cen, Hu Yu, Jingyun Liang, Shuning Chang, Zhihui Lin, Tao Feng, Pengwei Liu, Jiazheng Xing, et al. Lumos-1: On autoregressive video generation from a unified model perspective. *arXiv preprint arXiv:2507.08801*, 2025.

Kaiwen Zheng, Yongxin Chen, Hanzi Mao, Ming-Yu Liu, Jun Zhu, and Qinsheng Zhang. Masked diffusion models are secretly time-agnostic masked models and exploit inaccurate categorical sampling. In *ICLR*, 2025.

# APPENDIX

We will publish our code and pre-trained models to improve interpretability and assure reproducibility. Here, more implementation details, experiments and qualitative results are organized as follows:



## A USAGE OF LARGE LANGUAGE MODELS

We hereby declare that LLMs were used solely for the purpose of polishing this manuscript. There are no content generation, data interpretation, or intellectual contributions were derived from the LLMs. All ideas, analyses, and conclusions presented in this paper are the original work of the authors.

## B DISCUSSION

**Linearized metric path.** Shaul et al. (2025) introduces a general metric probability path and demonstrate their effectiveness on small-scale image/text data. FUDOKI (Wang et al., 2025b) further integrates this probability path into multimodal models, showing benefits for unified understanding and generation. However, these works do not address fundamental challenges arising in long-sequence visual generation, *e.g.*, the tailored design of metric path and spatiotemporal consistency. In this work, we introduce UDM, the first discrete framework that formulates video synthesis as iterative refinement over discrete spatiotemporal tokens. Specifically, Linearized Metric Path simplifies the design space of metric path, and Resolution-dependent Timestep Shifting enables stable long-sequence training, together addressing above two fundamental challenges that prior discrete methods could not handle.

**Resolution-dependent Timestep Shifting in the Discrete Space.** In continuous diffusion models, resolution-dependent timestep shifting is motivated by an explicit SNR analysis (Esser et al., 2024; Hoogeboom et al., 2023). Higher resolutions correspond to higher signal energy and requires stronger perturbations (lower SNR) at earlier timesteps to stabilize training and fully exploit the noise schedule. In contrast, prior discrete methods lack an analogous, well-defined SNR metric and thus can only import such schedules *heuristically* from continuous models. In this work, we approximate the SNR-like embedding distance along the proposed *Linearized Metric Path*. Under this interpretation, the resolution-dependent timestep shifting strategy arises naturally, rather than being post hoc tuned. To our knowledge, UDM is the first discrete framework that both *derives* and *systematically validates* a resolution-aware schedule, giving a conceptual basis to heuristics taken from continuous diffusion.

**Asynchronous Timestep Scheduling.** SkyReels-V2 (Chen et al., 2025a), Pusa (Liu et al., 2025b), FVDM (Liu et al., 2024) and UDM all relate to diffusion-forcing style schedulers that perform frame-wise perturbation and can support multitask generation. While prior works have extensively explored continuous timestep objectives, schedules, and conditioning, the corresponding design space in discrete methods remains unexplored. As summarized in Table 4, UDM differs from continuous models in several fundamental aspects. These differences prevent the direct reuse of continuous-space timestep designs. Our analysis (Fig.5 in Main text) shows that UDM needs remove timestep conditioning to achieve stable training and strong performance, revealing a fundamental difference from continuous diffusion methods. We believe that identifying this distinction provides useful

| Method | Noise | Loss | Scheduler | Timestep Conditioning |
|---|---|---|---|---|
| Continuous | Gaussian noise | MSE | Flow Matching | Yes |
| Discrete | Categorical noise | CE | Discrete Flow Matching | No |

Table 4: Comparison between continuous models and UDM in latent space, noise type, objective, and scheduling design.

insight for the community and may help accelerate progress on discrete video generation approaches.

## C Training and Sampling Details

**Algorithm 1 UDM** Training

**Require:** Predictor $p_\theta$, Steps $T$, Schedule $\beta_t$, Shift $\lambda$, Embedding $\mathbf{E} \in \mathbb{R}^{V \times d}$

**repeat**
  $x_1 \sim p_{\text{data}}$
  $t \sim \mathcal{U}(0, 1)$
  $\tilde{t} \leftarrow t/(t + \lambda(1 - t))$
  $x_{\tilde{t}} \sim p_{\tilde{t}|1}(\cdot|x_1)$
  $\mathcal{L} \leftarrow -\sum_{i=1}^{D} \log p_\theta(x_1^i \mid x_{\tilde{t}})$
  $\theta \leftarrow \theta - \eta \nabla_\theta \mathcal{L}$
**until** converged
**return** Trained predictor $p_\theta$

**Algorithm 2 UDM** Sampling

**Require:** Predictor $p_\theta$, Steps $T$, Schedule $\beta_t$, Shift $\lambda$, Embedding $\mathbf{E} \in \mathbb{R}^{V \times d}$

Sample $x_1 \sim \mathcal{U}(0, V)$, get $\mathbf{e}_1 = \mathbf{E}_{x_1}$
**for** $k = 1$ to $T$ **do**
  $t \leftarrow (k - 1)/T$
  $\tilde{t} \leftarrow t/(t + \lambda(1 - t))$
  $x_1 \sim p_\theta(\cdot|\mathbf{x}_{\tilde{t}})$
  $u_{\tilde{t}} \leftarrow u_t^\star(x, z|x_1)$
  $x_1 \leftarrow x_1 + dt \cdot u_{\tilde{t}}$
**end for**
**return** $x_1$ ▷ Generated discrete sample

**Training.** We encode images and videos into discrete latent tokens using the pre-trained tokenizer. For visual tokens, we adopt a DFM training objective based on the probability path. At each iteration, we randomly sample a timestep $t \in [0, 1]$ and use the metric path to obtain the noised tokens $x_t$. Text prompts are tokenized using the Qwen3 tokenizer and embedded into the same semantic space. We concatenate text embeddings and noised visual tokens into a unified sequence. The training objective is defined as the expected cross-entropy between the ground-truth visual token sequence and the model's predicted distribution. For the complete algorithmic workflow, please refer to Algorithm 1.

**Sampling.** This velocity field ensures that transitions occur only from state $z$ to state $x$ when $x$ is closer to $x_1$ than $z$, *i.e.*, $d(x, x_1) < d(z, x_1)$. Using the distance metric and the time-dependent factor $\beta_t$, the velocity guides the flow of particles in a manner that is both kinetic-optimal and aligned with the underlying geometry of the state space. We list the complete sampling process in Algorithm 2.

## D Implementation Details

**Training details.** UDM is trained on 128 A100 (40GB) GPUs. In all experiments, we use the AdamW optimizer (Loshchilov & Hutter, 2019) with $\beta_1 = 0.9$, $\beta_2 = 0.95$, weight decay of 0.05, and an initial learning rate of 1e-4. The learning rate employs cosine decay (Loshchilov & Hutter, 2017). We first pre-train text-to-image models and leverage their weights to initialize text-to-video models. Subsequently, following Chen et al. (2025a), we adapt full-sequence video diffusion models to diffusion forcing architectures by applying frame-wise noise schedules for autoregressive generation.

**Evaluation.** We evaluate text-to-image alignment using benchmarks DPG-Bench (Hu et al., 2024) and GenEval (Ghosh et al., 2024). Each image is generated from original or rewritten text prompts, with resolution determined by model type: 1024×1024 for image generation models to support high fidelity, and 512×320 for video generation models to effectively measure cross-modal generalization. We access text-to-video generation using VBench (Huang et al., 2024a) and image-to-video generation with VBench++ (Huang et al., 2024b), its comprehensive successor tailored for real-world scenarios. The videos, sized 49×512×320, are generated from rewritten prompts for text-to-video evaluation,

and from original text prompts with official cropped first-frame images for image-to-video evaluation. We apply classifier-free guidance (Ho & Salimans, 2022) with a scale value of 7.0 in all evaluations.

## E VIDEO EXTRAPOLATION EXPERIMENTS

As UDM is trained by applying independent noise levels to each frame, it naturally lends itself to video extrapolation via a sliding window. Specifically, new frames are generated sequentially, conditioned on the most recent 13 frames, thereby extending future predictions beyond the initial 49-frame context window. To effectively mitigate sampling errors in autoregressive video generation, we introduce a small amount of noise into historical frames by resampling them at timestep $t = 0.9$. Figure 7 presents the qualitative results for a video of 481 frames, where the initial text-to-video segment is extended through 12 extrapolation steps, producing videos up to 10× the original length.

Prompt: A woman walks down a Tokyo street filled with warm glowing neon and animated city signage.

Prompt: An extreme close-up of an gray-haired man with a beard in his 60s.

Figure 7: **Zero-shot video extrapolation.** We extend the 4-second text-to-video result to 40 seconds.

## F START-END FRAME CONTROL EXPERIMENTS

We evaluate UDM on the start-end frame control task, a specialized form of video generation to prevent future predictions from drifting. Concretely, we extract a sequence of frames from the video at 4-second intervals and place them sequentially at the beginning and the end of the context window. This setup enables the generation of a video featuring coherent motion of both objects and cameras, preserving spatial relationships throughout the scene. We present the qualitative results in Figure 8.

## G EXPERIMENTS ON THE EFFECT OF METRIC PATH LINEARITY

For the experiment setting, the left plot shows the L2 distance between the embeddings of noisy images (obtained by adding noise at different time steps $t$) and the embedding of the clean image, computed using 10k randomly selected images from the training set. The right plot follows the same experimental setting as the ablation study presented in the main text. We compute the Pearson correlation coefficients between the Euclidean distance and the timestep, which are -0.995, -0.921, -0.997, and -0.949. We found the choice of the probability path is significantly influenced by the values of $c$ and $\alpha$, and this in turn has a substantial impact on the model's performance. To determine optimal values for $c$ and $\alpha$, we draw inspiration from continuous diffusion model SD3 (Esser et al., 2024), where the relationship between $t$ and $d(x_t, x_1)$ demonstrates a strong linear correlation. This

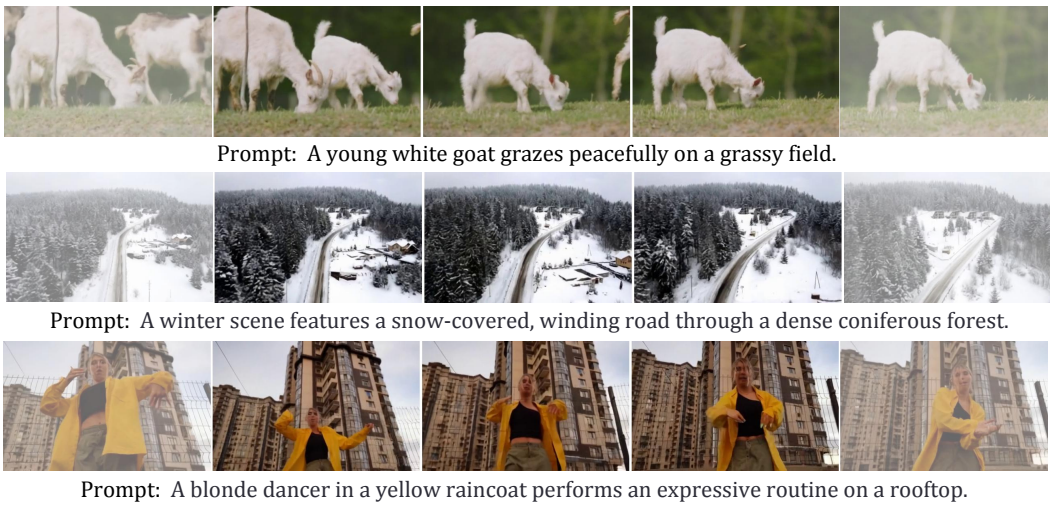
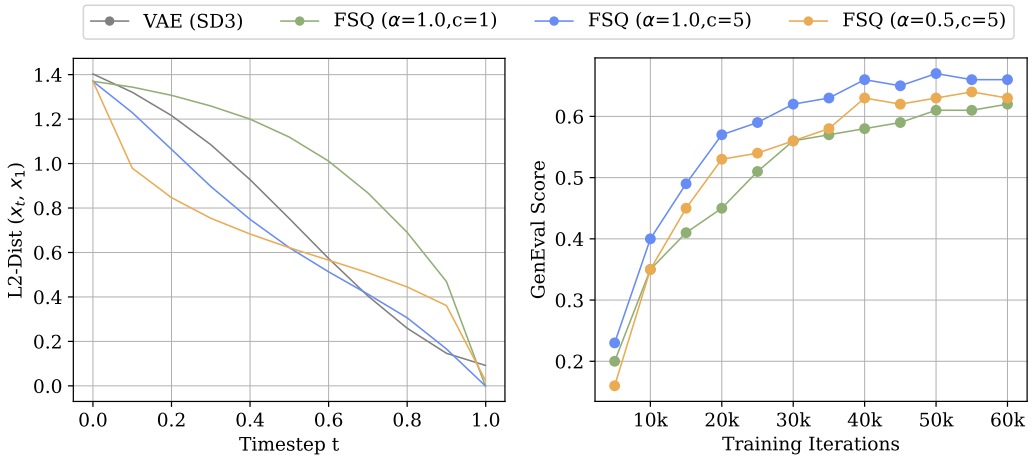



Figure 8: **Zero-shot start-end frame control.** The start-end frames are rendered with transparency.

insight guides our approach to calibrating $c$ and $\alpha$ to effectively reach the limits of model performance for different vision tokenizers.




(a) Linearized and non-linearized metric paths. (b) Text-to-image performance on GenEval.

Figure 9: **Sampling performance of different paths.** We evaluate the image samples at 256×256.

## H EXPERIMENTS OF MODEL SIZE FOR UNIFORM DIFFUSION

To study the scaling properties of UDM models, we train three variants that are initialized from Qwen3 models with 0.6B, 1.7B, and 4B parameters. Figure 10 compares the performance of different model sizes on DPG-Bench, GenEval, and VBench, with all models trained for the same epoch count as in Sec. 4.2. We find that increasing model size considerably enhances semantic performance across both text-to-image and text-to-video evaluations but does not significantly improve generation quality. This suggests that while larger models better capture high-level semantics and align more accurately with text prompts, the fidelity of the generated outputs may ultimately be constrained by the representation capacity of the discrete vision tokenizer.

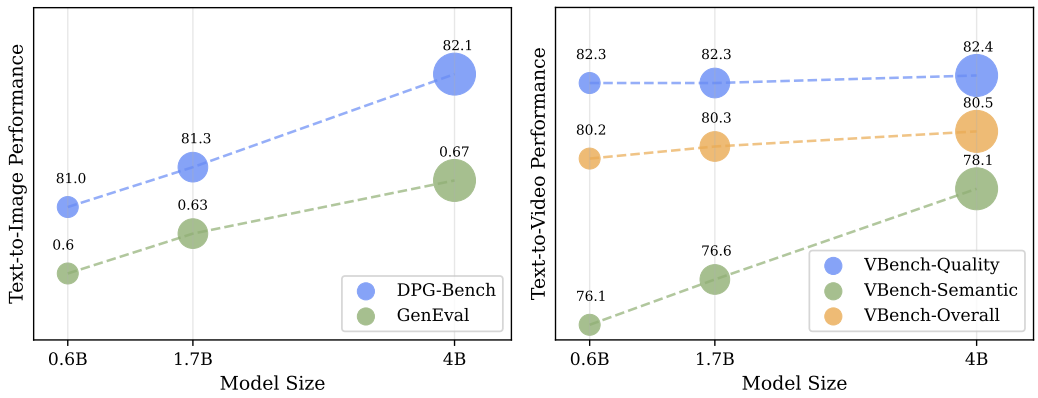



Figure 10: **Sampling performance of different model sizes.** All models are trained for the same epoch count as in the main experiments and evaluated on 256×256 images and 25×384×240 videos.

Table 5: **Comparison of different LLM backbones.**

| Model | #images | VisionTokenizer | Resolution | GenEval |
|---|---|---|---|---|
| Qwen3-0.6B | 24M | FSQ | 256×256 | 0.60 |
| Qwen3-1.7B | 24M | FSQ | 256×256 | 0.63 |
| Llama3.2-1B | 24M | FSQ | 256×256 | 0.61 |

## I Ablations on the impact of different LLM backbones

To assess sensitivity to the LLM backbones, we conduct an ablation in Table 5 comparing Qwen3-0.6B, Llama3.2-1B, and Qwen3-1.7B. All models share the same FSQ tokenizer, are trained on the same data for the same epoch count as in Sec. 4.2, and are evaluated on 256×256 images. GenEval improves monotonically from Qwen3-0.6B → Llama3.2-1B → Qwen3-1.7B, closely tracking model size. These results suggest that performance is primarily governed by model capacity rather than the specific LLM architecture, and our method is largely insensitive to the choice of Qwen versus other open-source backbones.

## J Inference speed comparison

To quantify the efficiency of our model, we report the average inference latency and compare UDM with representative continuous and discrete video generation baselines in Table 6. For a standard 49-frame setting, UDM-1.7B requires only 70s on 19K tokens, yielding a VBench-T2V score of 81.9, while a higher-resolution configuration reaches 82.4 with 180s latency. This places UDM on a more favorable speed–quality frontier than previous discrete models (e.g., Lumos1-3.6B and Emu3-8B) and comparable continuous diffusion models (e.g., CogVideoX-5B and StepVideo-30B).

Table 6: **Inference latency of UDM and prior video generation models.** UDM attains competitive VBench-T2V scores with lower or comparable latency than both discrete and continuous baselines. *Since the video version of Emu3-8B is not open-sourced, we estimate its latency by proportionally scaling from the officially released image model.

| **Model** | **Latent** | **Video Size** | **#Tokens** | **Latency (s)** | **VBench-T2V** |
|---|---|---|---|---|---|
| CogVideoX-5B | Continuous | 49 × 720 × 480 | 18K | 180s | 81.9 |
| StepVideo-30B | Continuous | 136 × 992 × 544 | 36K | 900s | 81.8 |
| Lumos1-3.6B | Discrete | 25 × 448 × 256 | 13K | 180s | 78.3 |
| Emu3-8B* | Discrete | 49 × 512 × 512 | 53K | 1700s | 81.0 |
| UDM-1.7B | Discrete | 49 × 240 × 384 | 19K | **70s** | 81.9 |
| UDM-1.7B | Discrete | 49 × 512 × 320 | 34K | 180s | **82.4** |

## K FORMAL DEFINITION AND OPTIMIZATION OF LINEARIZED METRIC PATH

**Motivation.** In the uniform discrete diffusion framework, the probability path $p_t(x \mid x_1)$ governs how the model observes visual tokens at different levels of corruption. A desirable property of this path is that the expected corruption strength increases *uniformly* over time. Intuitively, maintaining such linearity ensures uniform coverage across $t \in [0, 1]$, providing unbiased supervision across noise levels, stabilizing training from scratch, and yielding smoother refinement trajectories.

**Definition of Metric-Linearity.** Let $x_1$ denote a clean token sequence, $x_t \sim p_t(\cdot \mid x_1; \theta)$ the corrupted sample at time $t$, and $E(\cdot)$ the tokenizer embedding. The metric-induced probability path used in UDM is parameterized by $\theta = (c, \alpha)$:

$$\beta_t(\theta) = c \left( \frac{t}{1-t} \right)^{\alpha}, \tag{7}$$

$$p_t(x|x_1; \theta) = \mathrm{softmax}(-\beta_t(\theta)\, d(x, x_1)), \tag{8}$$

where $d(\cdot, \cdot)$ denotes the embedding-space distance induced by the tokenizer codebook. We define the expected corruption trajectory:

$$f(t) \;=\; \mathbb{E}_{x_1 \sim q,\; x_t \sim p_{t|1}(\cdot|x_1)} \Big[ \, \|E(x_t) - E(x_1)\|_2^2 \, \Big]. \tag{9}$$

which measures the expected deviation from clean samples at time $t$. We call the path *metric-linear* if $f_\theta(t)$ admits a close linear approximation in $t$; i.e., if the corruption level progresses at an approximately constant rate.

**Formulating Metric-Path Design as an Optimization Problem.** Prior approaches (Shaul et al., 2025) often rely on costly trial-and-error—training full models under different schedules and selecting the best performing one. Under the metric-linear hypothesis, UDM instead formulates the schedule selection as a bi-level optimization problem. For a discrete grid $\{t_k\}_{k=1}^K \subset [0, 1]$, let $(a_\theta, b_\theta)$ denote the best affine approximation to $f_\theta(t)$:

$$(a_\theta, b_\theta) \;=\; \arg\min_{a,b} \sum_{k=1}^{K} \left( f_\theta(t_k) - (a t_k + b) \right)^2. \tag{10}$$

The optimal schedule parameters are then obtained by minimizing the deviation from linearity:

$$\theta^\star \;=\; \arg\min_{\theta=(c,\alpha)} \sum_{k=1}^{K} \left( f_\theta(t_k) - (a_\theta t_k + b_\theta) \right)^2. \tag{11}$$

Equivalently, equation 11 can be expressed as maximizing the Pearson correlation between $f_\theta(t)$ and time $t$, providing a principled criterion for selecting metric paths without training generative models.

**Practical Solution.** Because $\theta = (c, \alpha)$ is two-dimensional and the inner regression admits a closed-form solution, equation 11 can be solved efficiently. We approximate $f_\theta(t)$ using a set of 10k randomly selected clean images from the training distribution and a finite grid of timesteps. A deterministic coarse-to-fine search over $\theta$ yields schedules whose corruption trajectories exhibit near-perfect linearity (Pearson correlation $\approx 0.99$). This procedure produces stable, well-behaved probability paths that significantly reduce path-design overhead compared with previous methods.