Lumina-Video: Efficient and Flexible Video Generation with Multi-scale Next-DiT

Dongyang Liu [*1 2] **Shicheng Li** [*2] **Yutong Liu** [*2] **Zhen Li** [*1] **Kai Wang** [*2] **Xinyue Li** [*2] **Qi Qin** [2] **Yufei Liu** [2] **Yi Xin** [2] **Zhongyu Li** [2 3] **Bin Fu** [2] **Chenyang Si** [2] **Yuewen Cao** [2] **Conghui He** [2] **Ziwei Liu** [2] **Yu Qiao** [2] **Qibin Hou** [3] **Hongsheng Li** [¶ 1 2] **Peng Gao** [¶ ♦ 2]

Abstract

Recent advancements have established Diffusion Transformers (DiTs) as a dominant framework in generative modeling. Building on this success, Lumina-Next achieves exceptional performance in the generation of photorealistic images with Next-DiT. However, its potential for video generation remains largely untapped, with significant challenges in modeling the spatiotemporal complexity inherent to video data. To address this, we introduce Lumina-Video, a framework that leverages the strengths of Next-DiT while introducing tailored solutions for video synthesis. Lumina-Video incorporates a Multi-scale Next-DiT architecture, which jointly learns multiple patchifications to enhance both efficiency and flexibility. By incorporating the motion score as an explicit condition, Lumina-Video also enables direct control of generated videos' dynamic degree. Combined with a progressive training scheme with increasingly higher resolution and FPS, and a multi-source training scheme with mixed natural and synthetic data, Lumina-Video achieves remarkable aesthetic quality and motion smoothness at high training and inference efficiency. We additionally propose Lumina-V2A, a video-to-audio model based on Next-DiT, to create synchronized sounds for generated videos. Codes are released at https://www.github.com/Alpha-VLLM/Lumina-Video.

1. Introduction

The field of generative modeling has witnessed significant advancements in recent years, with Diffusion Transformers (DiTs) emerging as a powerful paradigm for creating high-quality photorealistic content (Peebles & Xie, 2023; Esser et al., 2024; OpenAI, 2024). One notable innovation is Next-DiT (Zhuo et al., 2024), an improved version of flow-based DiT that has shown strong performance in image generation. By combining architectural enhancements such as 3D RoPE for superior spatiotemporal representation, sandwich normalization for stabilized training, and grouped-query attention for efficient attention computation, Next-DiT has achieved remarkable success. The model consistently produces images that are not only visually compelling but also exhibit high diversity and fine-grained details. However, despite its advancements in image synthesis, the potential of Next-DiT for video generation remains underexplored.

Video generation poses unique challenges that go beyond those encountered in image generation. The inherent complexity of modeling both spatial and temporal dimensions in a coherent manner introduces significant computational and architectural challenges. While Next-DiT can be adapted for video tasks, its current design is not specifically tailored for the spatiotemporal intricacies of video data, leading to an excessive number of video tokens and low computational efficiency. These limitations underscore the need for a tailored approach that fully leverages the capabilities of Next-DiT while addressing the unique demands of video synthesis.

To bridge this gap, we introduce Lumina-Video, a novel framework that excels at generating high-quality videos by building upon the strengths of Next-DiT. At the core of Lumina-Video is Multi-scale Next-DiT, an extension of Next-DiT into a multi-scale architecture by introducing multiple patch sizes that share a common DiT backbone and are trained jointly in a unified manner. This straightforward yet elegant approach allows the model to learn video structures across different computational budgets simultaneously. By strategically allocating different patchifications to various sampling timesteps, Lumina-Video achieves notable improvements in inference efficiency with only a minor sacrifice in quality. This design also enables users to dynamically adjust the computational cost based on resource constraints and specific requirements, offering greater flexibility during inference. Considering the importance of motion in

[*]Equal Contribution [¶]Corresponding Authors [♦]Project Lead [1]The Chinese University of Hong Kong [2]Shanghai AI Laboratory [3]Nankai University. Correspondence to: Peng Gao <gaopeng@pjlab.org.cn>, Hongsheng Li <hsli@ee.cuhk.edu.hk>.

Prompt: A massive explosion on the surface of the earth

Prompt: A panda drinking coffee in a cafe in Paris, black and white

Prompt: A red-haired child in glasses, dressed in a brown shirt and holding a large shoulder bag, looks off-camera with curiosity or anticipation. The softly blurred background, likely a living room or study with a bookshelf or bulletin board, adds a homely feel to the scene.

Prompt: In the video, a man is standing in an indoor setting, possibly a lounge or a party venue, given the presence of greenery in the background. He is dressed in a vibrant, patterned shirt with a collar and a tie, layered with a necklace and a pendant. The man is holding a glass, which appears to contain a beverage, in his right hand, and he seems to be engaged in conversation or addressing an audience. His expression is animated, and his mouth is open as if he is speaking or reacting to something. The overall atmosphere suggests a social or celebratory event.

Figure 1. Lumina-Video demonstrates a strong ability to generate high-quality videos with rich details and remarkable temporal coherence, accurately following both simple and detailed text prompts.

videos, we additionally derive motion scores from optical flow and incorporate them as an extra conditioning input to the DiT. By designing a systematic strategy that separately manipulates the motion conditioning of positive and negative classifier-free guidance (CFG) (Ho & Salimans, 2022) samples, Lumina-Video provides an effective interface for controlling the extent of dynamics in generated videos. We further refine the training strategy by progressively training the model on videos with increasing spatiotemporal resolutions to improve training efficiency, leveraging joint image-video training to enhance frame quality and text comprehension, and incorporating multi-source training to fully utilize diverse real and synthetic data sources. These designs enable Lumina-Video to seamlessly tackle the challenging video generation task across a wide range of scenarios.

Our contributions establish Lumina-Video as a new solution for video generation, offering researchers and practitioners a powerful and flexible tool for creating video content. By fully unleashing the potential of Multi-Scale Next-DiT, Lumina-Video is capable of generating high-fidelity videos of varying resolution, excelling in both quantitative metrics and visual quality. In addition, we design a Lumina-V2A, a Next-DiT-based video-to-audio framework to bring generated silent videos to real life with synchronized sounds. Furthermore, in line with our commitment to democratizing access to advanced video generation technologies, we open-source our training framework and model parameters, empowering the research community to explore, extend, and deploy Lumina-Video across a diverse range of applications. Through this work, we aim to catalyze future innovations in video generation and pave the way for broader adoption of generative modeling techniques.

2. Related Work

2.1. Video Generation

The field of video generation has evolved rapidly, combining advances in image synthesis with the additional complexity of temporal consistency across video frames. Early models predominantly relied on GANs for video generation (Saito et al., 2017; Skorokhodov et al., 2022; Tulyakov et al., 2018), which were capable of producing videos with rich details

but often suffered from unstable training and mode collapse. More recently, following breakthroughs in image generation and language modeling, the landscape of video generation has expanded to include diverse paradigms such as masked modeling (Villegas et al., 2023; Yu et al., 2023; 2024), autoregressive modeling (Hong et al., 2023; Kondratyuk et al., 2024; Jin et al., 2024b; Liu et al., 2024), and diffusion models (Ge et al., 2023; Gupta et al., 2024; Blattmann et al., 2023; Yang et al., 2024). Among these, diffusion models have emerged as the dominant approach due to their ability to produce high-quality videos with exceptional temporal consistency, as evidenced by their adoption in state-of-the-art proprietary systems like Sora (OpenAI, 2024). However, the high computational cost of training diffusion-based text-to-video models remains a significant obstacle. In this work, we address this challenge by introducing Lumina-Video, a multi-scale diffusion-based framework, achieving remarkable results in text-to-video generation with reduced computational burden.

2.2. Transformer-based Diffusion Models

Early diffusion models (Ho et al., 2020; Song et al., 2021) primarily relied on convolution-based U-Nets (Ronneberger et al., 2015). However, the rise of transformer architectures has revolutionized computer vision by achieving state-of-the-art performance across various tasks (Dosovitskiy et al., 2021; Carion et al., 2020; He et al., 2022). Building on this success, diffusion transformers like DiT (Peebles & Xie, 2023) and UViT (Bao et al., 2023) successfully adapted transformer architectures for visual generation, significantly advancing the field. This approach has since become the dominant paradigm in diffusion-based models, as demonstrated by the wide adoption in both open-source models and commercial systems (Esser et al., 2024; Ma et al., 2024b; Chen et al., 2024c;b; OpenAI, 2024). Recent work has also explored flow matching as an alternative to standard diffusion processes, offering improved efficiency and flexibility (Lipman et al., 2023; 2024; Liu et al., 2023b; Ma et al., 2024a). Notably, models like Lumina-T2X (Gao et al., 2024) and Lumina-Next (Zhuo et al., 2024) have refined the core components of flow-based diffusion transformers, achieving remarkable performance in image synthesis. Our work builds on these advances, extending the framework to video generation. By incorporating multi-scale learning and motion-aware conditioning, we adapt and enhance diffusion transformers to efficiently capture the complex spatiotemporal dynamics of videos.

2.3. Multi-scale Learning in Computer Vision

The concept of multi-scale processing has long been fundamental in computer vision (Koenderink, 2004; Burt & Adelson, 1983; Adelson et al., 1984). With the advent of deep learning, the idea of multi-scale processing has been revitalized, yielding significant benefits for tasks that demand both high-level semantics and low-level details (Lin et al., 2017; Chen et al., 2018; Eigen & Fergus, 2015; Pang et al., 2020; Chen et al., 2016; Cai et al., 2016; Gu et al., 2022; Fan et al., 2021; Liu et al., 2021; Zhang et al., 2020). In the context of visual generation, recent advances have embraced multi-scale architectures to enhance generative processes (Skorokhodov et al., 2024; Zhang et al., 2024a; Gu et al., 2024; Zheng et al., 2024a; Jin et al., 2024a). Building on these foundational works, we introduce a novel Multi-scale Next-DiT architecture for video generation. Our method extends the multi-scale paradigm to the spatiotemporal domain, learning video structures across multiple levels of detail with varying patch sizes. This design not only ensures efficient training but also excels in generating high-quality, temporally coherent videos.

3. Lumina-Video

In this section, we elaborate on the multi-patch method and the explicit injection of motion motioning. An introduction to the preliminaries, including the VAE, the text encoder, the Next-DiT architecture, and the loss function, is deferred to Sec.A in the appendix. A graphical illustration of the overall architecture is provided in Fig.2. An extra video-to-audio extension is introduced in Sec. D.

3.1. Multi-scale Next-DiT

In video generation based on diffusion transformers, the number of tokens processed by the transformer plays a pivotal role in determining the computational cost and training efficiency. More tokens allows the DiT to capture more fine-grained details, while it involves increased computational cost and degrades the efficiency

In this work, we propose Multi-scale Next-DiT, a novel architecture that incorporates multiple pairs of patchify and unpatchify layers trained in a unified manner. This architecture enables a systematic analysis of the impact of token quantities on the denoising process and demonstrates improved efficiency. By leveraging multiple scales, the model enables high efficiency by properly combining multiple scales in one complete denoising process. Moreover, our approach offers great flexibility that allows the model to adapt dynamically to diverse requirements in the inference stage. This adaptability advances the development and deployment of text-to-video models in multiple practical scenarios.

3.1.1. Multi-scale Patchification

In DiT-based T2V models, given a fixed VAE, the number of tokens is determined by two operations: patchify and unpatchify. The patchify layer converts the noised latent representation $\mathbf{z}_t \in \mathbb{R}^{T \times H \times W \times C}$ into a sequence of to-

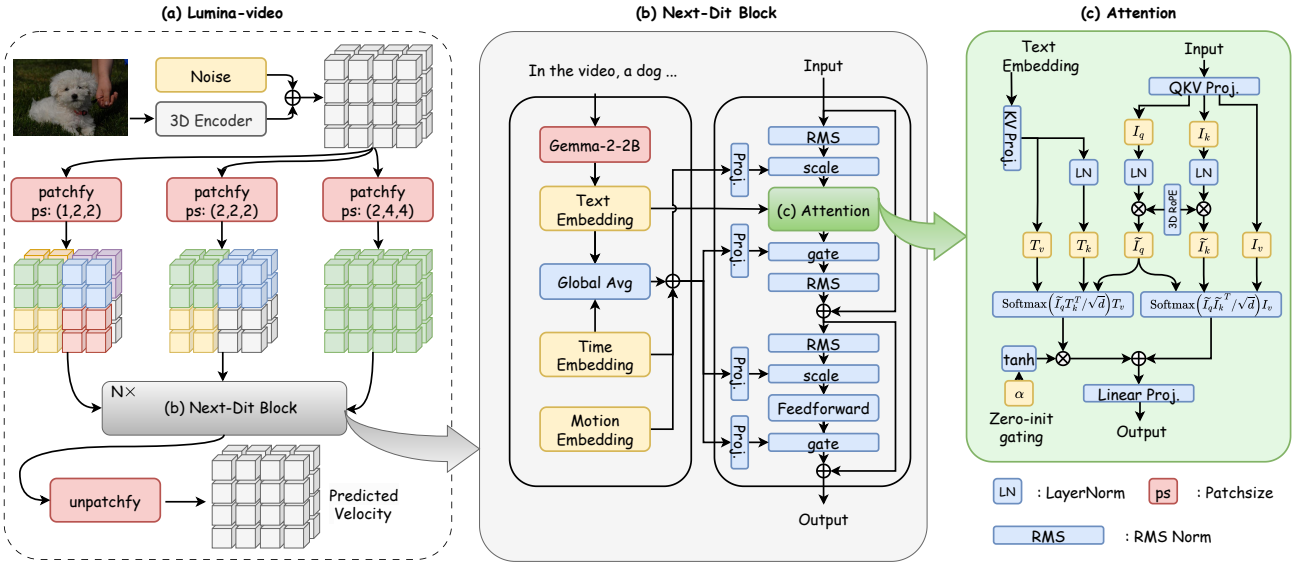

Figure 2. Architecture of Lumina-Video with Multi-scale Next-DiT and Motion Conditioning.

kens via a linear transformation before undergoing the DiT blocks, where the number of tokens can be calculated as

$$N = \frac{T \times H \times W}{p_t \times p_h \times p_w} \tag{1}$$

(p_t, p_h, p_w) denotes the patch size, a critical hyperparameter controlling the granularity of the representation. After the DiT blocks, the output tokens are linearly projected and reshaped back into the original shape by an unpatchify layer.

To endow the model with the ability to flexibly handle different levels of granularity based on varying computational requirements, we introduce the core component of Multi-scale Next-DiT: multi-scale patchification. Instead of using a single patch size, we instantiate multiple pairs of patchify and unpatchify layers with different spatio-temporal patch sizes, each corresponding to a distinct scale. Specifically, we employ a hierarchy of M patch sizes $\{P^i = (p_t^i, p_h^i, p_w^i) | i = 1, \cdots, M\}$, where $p_t^i \leq p_t^{i+1}$, $p_w^i \leq p_w^{i+1}$, $p_h^i \leq p_h^{i+1}$. Patchifying with a larger patch size leads to greater computation reduction, while patchifying with a smaller patch size preserves finer details in the latent representations. This allows us to dynamically adjust the level of abstraction used according to our demand.

Note that in Multi-scale Next-DiT, all patch sizes share the same DiT backbone. This design minimizes parameter count and memory overhead while facilitating knowledge sharing across scales.

3.1.2. Analysis

The unified training with the shared backbone and multiple patchifications provides a novel interface to investigate how different patch sizes affect the quality of denoising prediction, which can be directly reflected by loss magnitudes.

To do so, we evaluate three sets of patch sizes and observe their behavior across timesteps by dividing the denoising process uniformly into 20 time windows. We then analyze the training loss at various timesteps for different patch sizes. The three patch sizes correspond to different scales of spatiotemporal resolution: small, medium, and large.

Figure 3 shows the behavior of the loss curves across different patch sizes in 5 uniformly selected t spans, while the complete 20-span visualization is provided in Sec.C.2. In the very early stages of denoising ($0 \leq t \leq 0.1$), the loss curves for all three patch sizes overlap, indicating comparable prediction quality. As t increases, the loss curve for the smallest scale begins to deviate first, posing an obvious and stable gap compared to smaller patches. Similarly, as t grows even larger (say, after 0.4), the loss curve for the medium patch also deviates from that of the small patch, implying diverged prediction qualities. These findings validate previous assumptions about the varying nature of tasks performed at different timesteps: early stages of denoising focus on capturing the global structure, where smaller scales are sufficient to predict velocity accurately; as the process progresses, finer details become crucial, necessitating denoising on larger scales.

This empirical evidence justifies employing a hierarchical generation process by gradually increasing the scale throughout the denoising process, which reduces computational costs while maintaining quality.

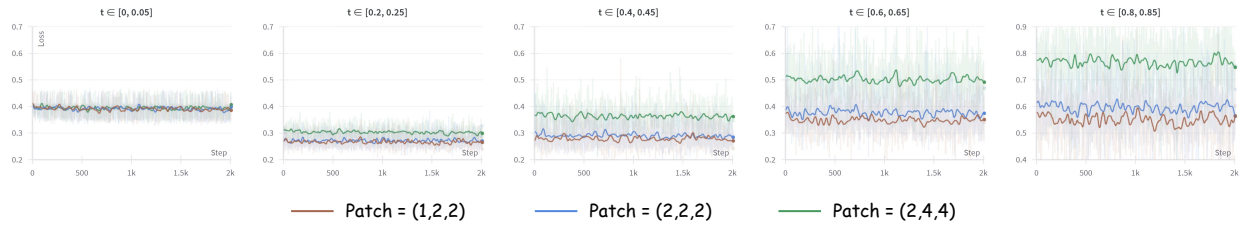

Figure 3. Loss curves for different patch sizes at different denoising timesteps. See Sec. C.2 for the complete figure.

Algorithm 1 Training of Multi-scale Next-DiT

Input: Model $\mathbf{u}^\theta$, dataset $\mathcal{D}$, batch size B, patch sizes $\mathcal{P} = \{P_i\}_{i=1}^M$, timeshift values $\{\alpha_i\}_{i=1}^M$, number of training steps T, learning rate η
for $iteration = 1$ **to** T **do**
  for $k = 1$ **to** M **do**
    Sample a batch of B samples from $\mathcal{D}$
    Sample t' uniformly from $[0, 1]$
    Compute rescaled time t with Eq. 2 and α_k
    Compute flow matching loss $\mathcal{L}_k$ with P_k
    Compute gradients $\nabla_\theta \mathcal{L}_k$ and accumulate
  end for
  Update model parameters θ with ∇_θ
  Zero out accumulated gradients: $\nabla_\theta \mathcal{L} \leftarrow 0$
end for

3.1.3. Training: Scale-aware Timestep Shifting

Based on the analysis in Section 3.1.2, we observe that larger patch sizes are more suitable for the early stages of denoising with a focus on capturing broad structures. In contrast, smaller patch sizes are more effective in the later stages, where the model needs to capture finer details as the noise diminishes. This observation suggests that applying different timestep sampling schedules to the training of different patch sizes can lead to more efficient training.

Rather than dividing the trajectory into discrete windows, which limits the interaction between different patch sizes across timesteps and hinders the sharing of knowledge between the various scales, we allow the timestep to be sampled from the entire trajectory $[0, 1]$ for all patch sizes. Inspired by Stable Diffusion 3 (Esser et al., 2024), we then customize the training resources allocation within different scales by assigning different time shift factors. Specifically, for each patch size P_i, we define a shift value α_i, which determines how much the sampled timestep is shifted toward the start of the trajectory. During training patch size P_i, a timestep t' is uniformly sampled from the interval $[0, 1]$ for each sample. The timestep t' is then mapped to the actual timestep t using the following formula:

$$t = \frac{t'}{t' + \alpha_i - \alpha_i t'} \tag{2}$$

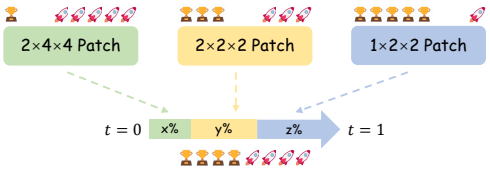

Figure 4. Multi-scale Patchification allows Lumina-Video to perform flexible multi-stage denoising during inference, leading to a better tradeoff between quality and efficiency.

For smaller patch sizes, we assign a smaller shift value to increase the likelihood of sampling larger timesteps, (i.e., later stages of denoising), where finer details are more important. Conversely, for larger patch sizes, we assign a larger shift value, which increases the probability of sampling smaller timesteps, where coarse structures are more relevant.

This shift-based allocation effectively tilts the training resources, ensuring that the most advantageous intervals for each patch size, balancing quality and efficiency, are sampled more frequently, thereby maximizing improvements in practical inference. Furthermore, it allows the model to more effectively share knowledge across stages and patch sizes and learn a unified representation. We summarize the training process in algorithm 1.

3.1.4. Flexible Multi-stage Inference

Training Multi-scale Next-DiT across multiple scales unlocks significant flexibility during inference. This flexibility is particularly valuable for resource-constrained scenarios, high-throughput requirements, or rapid prototyping when adjusting inference hyperparameters. Following the analysis in Sec.3.1.2, we propose a multi-stage denoising strategy: using smaller scales during early timesteps to determine the video's general structure, followed by refinement with increasingly larger scales in later stages. As illustrated in Figure 4, this approach enjoys the advantage of reducing computation with minor degradation in quality.

3.2. Explicit Control over Motion Condition

Text-to-video models often exhibit overly static behavior, indicating the need for an explicit mechanism for controlling the intensity of motion in the generated videos. In Lumina-

Table 1. Comparison on VBench. Proprietary models and open-source models are listed separately for better comparison.

Model	Param	Total Score (%)	Quality Score (%)	Semantic Score (%)	Motion Smoothness (%)	Dynamic Degree (%)
Proprietary Models						
Pika-1.0 (Labs, 2024)	-	80.69	82.92	71.77	99.50	47.50
Kling (Kuaishou, 2024)	-	81.85	83.39	75.68	99.40	46.94
Vidu (VIDU, 2025)	-	81.89	83.85	74.04	97.71	82.64
Gen-3 (Runway Research, 2024)	-	82.32	84.11	75.17	99.23	60.14
Luma (LumaLab, 2024)	-	83.61	83.47	84.17	99.35	44.26
Sora (OpenAI, 2024)	-	84.28	85.51	79.35	98.74	79.91
Open-Source Models						
OpenSora Plan V1.3 (Lab & etc., 2024)	2.7B	77.23	80.14	65.62	99.05	30.28
OpenSora V1.2 (8s) (Zheng et al., 2024b)	1.1B	79.76	81.35	73.39	98.50	42.39
VideoCrafter 2.0 (Chen et al., 2024a)	1.4B	80.44	82.20	73.42	97.73	42.50
Allegro (Zhou et al., 2024)	3B	81.09	83.12	72.98	98.82	55.00
CogVideoX (Yang et al., 2024)	5B	81.61	82.75	77.04	96.92	70.97
Pyramid Flow (Jin et al., 2024a)	2B	81.72	84.74	69.62	99.12	64.63
CogVideoX 1.5 (Yang et al., 2024)	5B	82.17	82.78	79.76	98.31	50.93
Vchitect 2.0 (Fan et al., 2025)	2B	82.24	83.54	77.06	98.98	63.89
HunyuanVideo (Kong et al., 2024)	13B	83.24	85.09	75.82	98.99	70.83
Lumina-Video (Single Scale)	2B	82.99	83.92	79.27	98.90	67.13
Lumina-Video (Multi Scale)	2B	82.94	84.08	78.39	98.92	71.76

Video, we introduce a motion conditioning mechanism that allows for direct control over motion characteristics.

Specifically, we condition the denoising process on a motion score in the same way as it is conditioned on the timestep, shown in Fig.2(b). During training, we calculate this motion score as the average of the magnitude of the optical flow using UniMatch (Yang et al., 2023), an off-the-shelf optical flow model. By conditioning the denoising process on this motion score, the model learns to generate videos with an aligned extent of dynamics. As validated in Sec. 5.2.2, by reasonably manipulating the motion conditioning for the positive and negative classifier-free guidance (CFG) samples, the generated dynamic degree can be adjusted effectively and reliably. Moreover, we introduce stochasticity by randomly dropping the motion condition with a probability $p = 0.4$ during training to handle situations where the user may not want explicit control over the motion score. This enables the model to adjust the intensity of motion based on the text prompt alone when no motion score is provided.

4. Training

We adopt a mixture of multiple training strategies:

Progressive Training has been widely recognized as an effective and efficient approach for training large-scale visual generative models (Liu et al., 2024; Gao et al., 2024; Yang et al., 2024; Lab & etc., 2024; Polyak et al., 2024a). Following this paradigm, Lumina-Video employs a 4-stage training process, beginning with text-to-image training in the first stage and transitioning to joint text-to-image/video training in the subsequent three stages. Each stage is characterized by a predefined spatial area while allowing variable aspect ratios, ensuring that images and video frames are resized to resolutions close to the specified area while preserving their original aspect ratios. Additionally, each stage is defined with a specific frames-per-second (FPS) value. A maximum clip duration of 4s is applied throughout the entire training.

In **Stage 1**, the model is trained on pure image data at the resolution of 256 pixels. By rapidly processing a large volume of image-text pairs at high throughput, the model quickly captures the general composition and distribution of visual data, and establishes broad associations between language terms and visual concepts. We use the same image dataset as Lumina-Next (Zhuo et al., 2024), which shares the similar distribution as JourneyDB (Sun et al., 2024)

In **Stage 2**, we incorporate the 10M official subset of the large-scale video dataset Panda (Chen et al., 2024d). The spatial resolution remains at 256 pixels, and video frames are extracted at a target FPS of 8. **Stage 3** raises the spatial resolution to 512 pixels and the FPS to 16, and training is conducted on a mixture of data from OpenVid (Nan et al., 2024), Open-Sora-Plan (Lab & etc., 2024), and 300k in-house video samples, which consist of diverse real and synthetic data from various sources. Finally, in **Stage 4**, the spatial resolution increases to 960 pixels, and the FPS is elevated to 24. The same dataset as Stage 3 but filtered with FPS and resolution is used in this stage.

Image-Video Joint Training is employed across stages 2 to 4. Leveraging the concept richness and superior quality of image data, this joint training significantly enhances the model's capacity to understand a broader spectrum of concepts and improves frame-level quality.

Multi-Scale Traning is introduced since stage 2. For video,

we define three patch sizes: $P_1 = (1, 2, 2)$, $P_2 = (2, 2, 2)$, and $P_3 = (2, 4, 4)$. For image, we use the $P_1 = (1, 2, 2)$ patchification only. We observe that applying coarser patchifications with temporal compression during image training leads to unintended effects: as images are repeated to fulfill patchification requirements, the model interprets them as *silent videos*, resulting in generated samples with the interesting phenomenon of intra-patch silence and inter-patch dynamics. Scale-aware time-shift introduced in Sec.3.1.3 is applied throughout progressive training.

Multi-Source Traning denotes a novel **multi-system-prompt training → per-system prompt evaluation → best-subset fine-tuning** strategy. When training relatively large-scale models, the available data typically comes from diverse sources with varying distributions and quality levels. Ideally, we expect the model to learn from all available samples while ensuring that the generated content during inference aligns closely with the quality of the best subset of the training data. However, this objective is challenging, especially when the complexity of data composition makes it difficult for even experienced practitioners to pinpoint which subset qualifies as optimal.

To solve this problem, we introduce distinct system prompts tailored to each data source into progressive training, and prepend them to the image prompts, forming complete prompts for training. When performing random prompt dropping for CFG, we drop only the image-related prompts while retaining the system prompts to preserve source-specific context during training.

After completing the progressive training, we evaluate the model performance using different system prompts. This evaluation involves both subjective assessments and benchmark-based quantitative metrics. Our observations reveal that generation quality and stability exhibit significant variation across different system prompts.

Based on evaluation results, we conduct a final *best-subset fine-tuning* stage at a reduced learning rate on the most effective system prompt subset. A few hundred iterations suffice to significantly enhance the model's performance, aligning outputs with the subset's characteristics while preserving the broad generalization achieved in earlier stages.

Notably, we find that **synthetic data**, despite comprising only a small portion ($\sim$10%) of our in-house dataset, consistently achieves higher scores in per-system prompt evaluation. By checking the generated samples, those from synthetic system prompts also demonstrate greater stability. These findings highlight the effectiveness of synthetic data in video generation, likely due to its simpler distribution, which prevents the model from getting confused by the erratic variability of the real world. To our knowledge, this is the first work to validate synthetic data's utility for large-scale video generation models. We believe these insights will aid research groups with limited resources in developing stronger, more efficient foundational video models.

Table 2. Ablation study on patchification.

PATCH	TIME COST	QUALITY	SEMANTIC	TOTAL
(1, 2, 2)	1.00	83.92	79.27	82.99
(2, 2, 2)	0.36	83.50	78.33	82.47
(2, 4, 4)	0.07	82.47	77.77	81.53
COMBINED	0.34	84.08	78.39	82.94

5. Evaluation

After completing the four stages of progressive training, we select the final checkpoints from stages 3 and 4 and then perform best-subset fine-tuning at the corresponding resolution and frame rate (FPS). This process results in two final models: one with a spatial resolution of 512^2 at 16 FPS, and another with 960^2 at 24 FPS. For our quantitative and ablation experiments, we use the first model by default. Demo samples are provided in the supplementary zip file.

5.1. Comparison with Existing Methods

Evaluation Benchmark We quantitatively evaluate the performance of Lumina-Video on VBench (Huang et al., 2024), a comprehensive benchmark for text-to-video generation. VBench consists of 16 fine-grained metrics from two primary dimensions including video quality (depicted by quality score) and video-text alignment (depicted by semantic score). During inference, we uniformly sample 70 timesteps and apply a shifting value $\alpha = 8.0$.

We present the quantitative results on VBench in Table 1, comparing Lumina-Video against both proprietary and open-source models. The results demonstrate that Lumina-Video is highly competitive overall, performing well in generating high-quality videos while effectively understanding and following user prompts. Detailed results for individual metrics are provided in Table 4 in the appendix.

5.2. Ablation Study

5.2.1. MULTI-SCALE PATCHIFICATION

Our experimental results in Tab. 2 and Fig. 5 show that using the smallest patch size throughout the entire process yields the highest overall performance, while larger patch sizes reduce model performance but raise generation speed. Compared to using a single patch size, our patchification stitching-based inference achieves a better efficiency-quality balance. further supporting the relationship between prediction quality, patch size, and timestep illustrated in Fig. 3

Table 3. Impact of Motion Score. * means Dynamic Degree is excluded from Quality Score.

POSITIVE MOTION	NEGTIVE MOTION	DYNAMIC DEGREE	QUALITY SCORE*	SEMANTIC SCORE	TOTAL SCORE
4	4	45.37	86.11	79.28	82.24
8	8	53.70	86.18	78.88	82.72
4	2	72.78	84.11	79.10	82.41
8	2	85.89	83.51	78.36	82.63
4	2-[0.05-4]	67.13	85.31	79.27	82.99
8	2-[0.05-8]	83.33	84.58	78.55	83.28

Figure 5. Comparison of generated videos using different patchification strategies.

5.2.2. MOTION SCORE

Lumina-Video has introduced motion score as a micro-condition for controlling dynamics. However, as shown in Tab. 3, increasing motion conditioning fails to enhance dynamics if the conditioning for CFG negative sample rises simultaneously: setting motion to 4 yields low dynamics, while increasing it to 8 provides only modest improvement.

This suggests the possibility that high dynamics depends on the difference between positive and negative motion conditioning, not their absolute values. To verify this, we fixed the negative motion at 2 and observed higher dynamics with [4,2] compared to [4,4], and further improvement with [8,2]. These results confirm motion difference as an effective control for video dynamics.

However, increasing this difference degrades content quality, as reflected in quality and semantic scores. To balance dynamics and quality, we propose initially setting the negative motion to a low value (e.g., 2) and aligning it with positive motion after a threshold (e.g., 0.05). This approach, based on the assumption that motion structures form earlier than fine details in diffusion, achieves an optimal balance, as shown in rows 5–6 in Tab. 3. A visualization of motion condition's impact is shown in Fig. 6

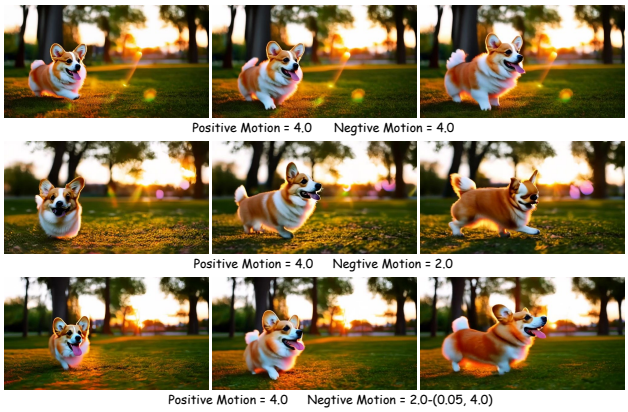

Figure 6. Comparison of generated videos using different positive and negative motion scores.

6. Conclusion & Futer Work

We present Lumina-Video, a novel framework designed to overcome the unique challenges of video generation by building on the successes of the Next-DiT architecture. Lumina-Video boosts efficiency by utilizing a Multi-scale Next-DiT design and allows direct control of the dynamic degree by explicit conditioning. Through a combination of strategies including progressive training and multi-source training, Lumina-Video demonstrates a powerful solution for generating high-fidelity videos with both spatial and temporal consistency. Additionally, the companion Lumina-V2A model further enhances real-world applicability through audio-visual synchronization.

Moving forward, we will focus on two key areas. First, multi-scale patchification shares the same motivation as dynamic neural networks, where simpler tasks require fewer resources. Insights from dynamic networks and network compression suggest that an organic compression across multiple dimensions (e.g., depth, width, tokens) generally offers better trade-offs than focusing on a single dimension, and we will explore this further. Second, while benchmarks show that Lumina-Video generates prompt-coherent videos meeting basic quality standards, with higher standards, gaps remain compared to commercial solutions in video aesthetics, complex motion synthesis, and artifact-free details. These challenges will drive our efforts in data curation, architecture refinement, and pipeline optimization.

7. Impact Statements

This paper discusses a video generation method. Video generation, as a promising technology, also comes with significant societal risks, many of which are common to generative models in general. These risks deserve careful consideration and detailed discussion. However, the potential impacts of our method are reflective of those of the broader video generation field. Therefore, we believe there is no need to specifically highlight any particular impact in this context.

References

Adelson, E. H., Burt, P. J., Anderson, C. H., Ogden, J. M., and Bergen, J. R. Pyramid methods in image processing. 1984. URL https://api.semanticscholar.org/CorpusID:1330580.

Bao, F., Nie, S., Xue, K., Cao, Y., Li, C., Su, H., and Zhu, J. All are worth words: A vit backbone for diffusion models. In CVPR, pp. 22669–22679. IEEE, 2023.

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.

Burt, P. J. and Adelson, E. H. The laplacian pyramid as a compact image code. IEEE Trans. Commun., 31:532–540, 1983. URL https://api.semanticscholar.org/CorpusID:8018433.

Cai, Z., Fan, Q., Feris, R. S., and Vasconcelos, N. A unified multi-scale deep convolutional neural network for fast object detection. In ECCV (4), volume 9908 of Lecture Notes in Computer Science, pp. 354–370. Springer, 2016.

Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A., and Zagoruyko, S. End-to-end object detection with transformers. In ECCV (1), volume 12346 of Lecture Notes in Computer Science, pp. 213–229. Springer, 2020.

Chen, H., Xie, W., Vedaldi, A., and Zisserman, A. Vggsound: A large-scale audio-visual dataset. In ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 721–725. IEEE, 2020.

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 CVPR, pp. 7310–7320. IEEE, 2024a.

Chen, J., Ge, C., Xie, E., Wu, Y., Yao, L., Ren, X., Wang, Z., Luo, P., Lu, H., and Li, Z. Pixart-Σ: Weak-to-strong training of diffusion transformer for 4k text-to-image generation. In ECCV (32), volume 15090 of Lecture Notes in Computer Science, pp. 74–91. Springer, 2024b.

Chen, J., Yu, J., Ge, C., Yao, L., Xie, E., Wang, Z., Kwok, J. T., Luo, P., Lu, H., and Li, Z. Pixart-α: Fast training of diffusion transformer for photorealistic text-to-image synthesis. In ICLR. OpenReview.net, 2024c.

Chen, L., Yang, Y., Wang, J., Xu, W., and Yuille, A. L. Attention to scale: Scale-aware semantic image segmentation. In CVPR, pp. 3640–3649. IEEE Computer Society, 2016.

Chen, L., Papandreou, G., Kokkinos, I., Murphy, K., and Yuille, A. L. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. IEEE Trans. Pattern Anal. Mach. Intell., 40(4):834–848, 2018.

Chen, T., Siarohin, A., Menapace, W., Deyneka, E., Chao, H., Jeon, B. E., Fang, Y., Lee, H., Ren, J., Yang, M., and Tulyakov, S. Panda-70m: Captioning 70m videos with multiple cross-modality teachers. In CVPR, pp. 13320–13331. IEEE, 2024d.

Chen, Z., Seetharaman, P., Russell, B., Nieto, O., Bourgin, D., Owens, A., and Salamon, J. Video-guided foley sound generation with multimodal controls. arXiv preprint arXiv:2411.17698, 2024e.

Cheng, H. K., Ishii, M., Hayakawa, A., Shibuya, T., Schwing, A., and Mitsufuji, Y. Taming multimodal joint training for high-quality video-to-audio synthesis. arXiv preprint arXiv:2412.15322, 2024.

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., and Houlsby, N. An image is worth 16x16 words: Transformers for image recognition at scale. In ICLR. OpenReview.net, 2021.

Eigen, D. and Fergus, R. Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture. In ICCV, pp. 2650–2658. IEEE Computer Society, 2015.

Esser, P., Kulal, S., Blattmann, A., Entezari, R., Müller, J., Saini, H., Levi, Y., Lorenz, D., Sauer, A., Boesel, F., Podell, D., Dockhorn, T., English, Z., and Rombach, R. Scaling rectified flow transformers for high-resolution image synthesis. In ICML. OpenReview.net, 2024.

Fan, H., Xiong, B., Mangalam, K., Li, Y., Yan, Z., Malik, J., and Feichtenhofer, C. Multiscale vision transformers. In ICCV, pp. 6804–6815. IEEE, 2021.

Fan, W., Si, C., Song, J., Yang, Z., He, Y., Zhuo, L., Huang, Z., Dong, Z., He, J., Pan, D., et al. Vchitect-2.0: Parallel transformer for scaling up video diffusion models. arXiv preprint arXiv:2501.08453, 2025.

Fang, A., Jose, A. M., Jain, A., Schmidt, L., Toshev, A., and Shankar, V. Data filtering networks. arXiv preprint arXiv:2309.17425, 2023.

Gao, P., Zhuo, L., Lin, Z., Liu, C., Chen, J., Du, R., Xie, E., Luo, X., Qiu, L., Zhang, Y., et al. Lumina-t2x: Transforming text into any modality, resolution, and duration via flow-based large diffusion transformers. arXiv preprint arXiv:2405.05945, 2024.

Ge, S., Nah, S., Liu, G., Poon, T., Tao, A., Catanzaro, B., Jacobs, D., Huang, J., Liu, M., and Balaji, Y. Preserve your own correlation: A noise prior for video diffusion models. In ICCV, pp. 22873–22884. IEEE, 2023.

Gu, J., Kwon, H., Wang, D., Ye, W., Li, M., Chen, Y., Lai, L., Chandra, V., and Pan, D. Z. Multi-scale high-resolution vision transformer for semantic segmentation. In CVPR, pp. 12084–12093. IEEE, 2022.

Gu, J., Zhai, S., Zhang, Y., Susskind, J. M., and Jaitly, N. Matryoshka diffusion models. In ICLR. OpenReview.net, 2024.

Gupta, A., Yu, L., Sohn, K., Gu, X., Hahn, M., Li, F., Essa, I., Jiang, L., and Lezama, J. Photorealistic video generation with diffusion models. In ECCV (79), volume 15137 of Lecture Notes in Computer Science, pp. 393–411. Springer, 2024.

He, K., Chen, X., Xie, S., Li, Y., Dollár, P., and Girshick, R. B. Masked autoencoders are scalable vision learners. In CVPR, pp. 15979–15988. IEEE, 2022.

Ho, J. and Salimans, T. Classifier-free diffusion guidance. arXiv preprint arXiv:2207.12598, 2022.

Ho, J., Jain, A., and Abbeel, P. Denoising diffusion probabilistic models. In NeurIPS, 2020.

Hong, W., Ding, M., Zheng, W., Liu, X., and Tang, J. Cogvideo: Large-scale pretraining for text-to-video generation via transformers. In ICLR. OpenReview.net, 2023.

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.

Iashin, V., Xie, W., Rahtu, E., and Zisserman, A. Synchformer: Efficient synchronization from sparse cues. In ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 5325–5329. IEEE, 2024.

Jeong, Y., Kim, Y., Chun, S., and Lee, J. Read, watch and scream! sound generation from text and video. arXiv preprint arXiv:2407.05551, 2024.

Jin, Y., Sun, Z., Li, N., Xu, K., Jiang, H., Zhuang, N., Huang, Q., Song, Y., Mu, Y., and Lin, Z. Pyramidal flow matching for efficient video generative modeling. arXiv preprint arXiv:2410.05954, 2024a.

Jin, Y., Sun, Z., Xu, K., Xu, K., Chen, L., Jiang, H., Huang, Q., Song, C., Liu, Y., Zhang, D., Song, Y., Gai, K., and Mu, Y. Video-lavit: Unified video-language pre-training with decoupled visual-motional tokenization. In ICML. OpenReview.net, 2024b.

Koenderink, J. J. The structure of images. Biological Cybernetics, 50:363–370, 2004. URL https://api.semanticscholar.org/CorpusID:206775432.

Kondratyuk, D., Yu, L., Gu, X., Lezama, J., Huang, J., Schindler, G., Hornung, R., Birodkar, V., Yan, J., Chiu, M., Somandepalli, K., Akbari, H., Alon, Y., Cheng, Y., Dillon, J. V., Gupta, A., Hahn, M., Hauth, A., Hendon, D., Martinez, A., Minnen, D., Sirotenko, M., Sohn, K., Yang, X., Adam, H., Yang, M., Essa, I., Wang, H., Ross, D. A., Seybold, B., and Jiang, L. Videopoet: A large language model for zero-shot video generation. In ICML. OpenReview.net, 2024.

Kong, J., Kim, J., and Bae, J. Hifi-gan: Generative adversarial networks for efficient and high fidelity speech synthesis. Advances in neural information processing systems, 33:17022–17033, 2020.

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.

Kuaishou. Kling ai, 2024. URL https://klingai.kuaishou.com. Accessed: 2024-11-26.

Lab, P.-Y. and etc., T. A. Open-sora-plan, April 2024. URL https://doi.org/10.5281/zenodo.10948109.

Labs, P. Pika, 2024. URL https://pika.art. Accessed: 2024-11-26.

Lee, J., Im, J., Kim, D., and Nam, J. Video-foley: Two-stage video-to-sound generation via temporal event condition for foley sound. arXiv preprint arXiv:2408.11915, 2024.

Lin, T., Dollár, P., Girshick, R. B., He, K., Hariharan, B., and Belongie, S. J. Feature pyramid networks for object detection. In CVPR, pp. 936–944. IEEE Computer Society, 2017.

Lipman, Y., Chen, R. T. Q., Ben-Hamu, H., Nickel, M., and Le, M. Flow matching for generative modeling. In ICLR. OpenReview.net, 2023.

Lipman, Y., Havasi, M., Holderrieth, P., Shaul, N., Le, M., Karrer, B., Chen, R. T., Lopez-Paz, D., Ben-Hamu, H., and Gat, I. Flow matching guide and code. arXiv preprint arXiv:2412.06264, 2024.

Liu, D., Zhao, S., Zhuo, L., Lin, W., Qiao, Y., Li, H., and Gao, P. Lumina-mgpt: Illuminate flexible photorealistic text-to-image generation with multimodal generative pretraining. arXiv preprint arXiv:2408.02657, 2024.

Liu, H., Chen, Z., Yuan, Y., Mei, X., Liu, X., Mandic, D., Wang, W., and Plumbley, M. D. Audioldm: Text-to-audio generation with latent diffusion models. arXiv preprint arXiv:2301.12503, 2023a.

Liu, X., Gong, C., and Liu, Q. Flow straight and fast: Learning to generate and transfer data with rectified flow. In ICLR. OpenReview.net, 2023b.

Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., Lin, S., and Guo, B. Swin transformer: Hierarchical vision transformer using shifted windows. In ICCV, pp. 9992–10002. IEEE, 2021.

Loshchilov, I. and Hutter, F. Decoupled weight decay regularization. arXiv preprint arXiv:1711.05101, 2017.

LumaLab. Dreammachine, 2024. URL https://lumalabs.ai/dream-machine. Accessed: 2024-11-26.

Luo, S., Yan, C., Hu, C., and Zhao, H. Diff-foley: Synchronized video-to-audio synthesis with latent diffusion models. Advances in Neural Information Processing Systems, 36, 2024.

Ma, N., Goldstein, M., Albergo, M. S., Boffi, N. M., Vanden-Eijnden, E., and Xie, S. Sit: Exploring flow and diffusion-based generative models with scalable interpolant transformers. In ECCV (77), volume 15135 of Lecture Notes in Computer Science, pp. 23–40. Springer, 2024a.

Ma, X., Wang, Y., Jia, G., Chen, X., Liu, Z., Li, Y.-F., Chen, C., and Qiao, Y. Latte: Latent diffusion transformer for video generation. arXiv preprint arXiv:2401.03048, 2024b.

Nan, K., Xie, R., Zhou, P., Fan, T., Yang, Z., Chen, Z., Li, X., Yang, J., and Tai, Y. Openvid-1m: A large-scale high-quality dataset for text-to-video generation. arXiv preprint arXiv:2407.02371, 2024.

OpenAI. Sora, 2024. URL https://openai.com/index/sora. Accessed: 2024-11-26.

Pang, Y., Zhao, X., Zhang, L., and Lu, H. Multi-scale interactive network for salient object detection. In CVPR, pp. 9410–9419. Computer Vision Foundation / IEEE, 2020.

Peebles, W. and Xie, S. Scalable diffusion models with transformers. In ICCV, pp. 4172–4182. IEEE, 2023.

Polyak, A., Zohar, A., Brown, A., Tjandra, A., Sinha, A., Lee, A., Vyas, A., Shi, B., Ma, C.-Y., Chuang, C.-Y., Yan, D., Choudhary, D., Wang, D., Sethi, G., Pang, G., Ma, H., Misra, I., Hou, J., Wang, J., Jagadeesh, K., Li, K., Zhang, L., Singh, M., Williamson, M., Le, M., Yu, M., Singh, M. K., Zhang, P., Vajda, P., Duval, Q., Girdhar, R., Sumbaly, R., Rambhatla, S. S., Tsai, S. S., Azadi, S., Datta, S., Chen, S., Bell, S., Ramaswamy, S., Sheynin, S., Bhattacharya, S., Motwani, S., Xu, T., Li, T., Hou, T., Hsu, W.-N., Yin, X., Dai, X., Taigman, Y., Luo, Y., Liu, Y.-C., Wu, Y.-C., Zhao, Y., Kirstain, Y., He, Z., He, Z., Pumarola, A., Thabet, A. K., Sanakoyeu, A., Mallya, A., Guo, B., Araya, B., Kerr, B., Wood, C., Liu, C., Peng, C., Vengertsev, D., Schonfeld, E., Blanchard, E., Juefei-Xu, F., Nord, F., Liang, J., Hoffman, J., Kohler, J., Fire, K., Sivakumar, K., Chen, L., Yu, L., Gao, L., Georgopoulos, M., Moritz, R., Sampson, S. K., Li, S., Parmeggiani, S., Fine, S., Fowler, T., Petrovic, V., and Du, Y. Movie gen: A cast of media foundation models. 2024a. URL https://api.semanticscholar.org/CorpusID:273403698.

Polyak, A., Zohar, A., Brown, A., Tjandra, A., Sinha, A., Lee, A., Vyas, A., Shi, B., Ma, C.-Y., Chuang, C.-Y., et al. Movie gen: A cast of media foundation models. arXiv preprint arXiv:2410.13720, 2024b.

Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al. Learning transferable visual models from natural language supervision. In International conference on machine learning, pp. 8748–8763. PMLR, 2021.

Ronneberger, O., Fischer, P., and Brox, T. U-net: Convolutional networks for biomedical image segmentation. In MICCAI (3), volume 9351 of Lecture Notes in Computer Science, pp. 234–241. Springer, 2015.

Runway Research. Gen-3 alpha. https://runwayml.com/research/introducing-gen-3, 2024. Accessed: 2025-01-29.

Saito, M., Matsumoto, E., and Saito, S. Temporal generative adversarial nets with singular value clipping. In ICCV, pp. 2849–2858. IEEE Computer Society, 2017.

Skorokhodov, I., Tulyakov, S., and Elhoseiny, M. Stylegan-v: A continuous video generator with the price, image quality and perks of stylegan2. In CVPR, pp. 3616–3626. IEEE, 2022.

Skorokhodov, I., Menapace, W., Siarohin, A., and Tulyakov, S. Hierarchical patch diffusion models for high-resolution video generation. In CVPR, pp. 7569–7579. IEEE, 2024.

Song, J., Meng, C., and Ermon, S. Denoising diffusion implicit models. In ICLR. OpenReview.net, 2021.

Sun, K., Pan, J., Ge, Y., Li, H., Duan, H., Wu, X., Zhang, R., Zhou, A., Qin, Z., Wang, Y., et al. Journeydb: A benchmark for generative image understanding. Advances in Neural Information Processing Systems, 36, 2024.

Team, G., Riviere, M., Pathak, S., Sessa, P. G., Hardin, C., Bhupatiraju, S., Hussenot, L., Mesnard, T., Shahriari, B., Ramé, A., et al. Gemma 2: Improving open language models at a practical size. arXiv preprint arXiv:2408.00118, 2024.

Tulyakov, S., Liu, M., Yang, X., and Kautz, J. Mocogan: Decomposing motion and content for video generation. In CVPR, pp. 1526–1535. Computer Vision Foundation / IEEE Computer Society, 2018.

VIDU. Vidu: Online video learning platform. https://www.vidu.com/zh, 2025. Accessed: 2025-01-29.

Villegas, R., Babaeizadeh, M., Kindermans, P., Moraldo, H., Zhang, H., Saffar, M. T., Castro, S., Kunze, J., and Erhan, D. Phenaki: Variable length video generation from open domain textual descriptions. In ICLR. OpenReview.net, 2023.

Wang, H., Ma, J., Pascual, S., Cartwright, R., and Cai, W. V2a-mapper: A lightweight solution for vision-to-audio generation by connecting foundation models. In Proceedings of the AAAI Conference on Artificial Intelligence, volume 38, pp. 15492–15501, 2024.

Wang, Y., Guo, W., Huang, R., Huang, J., Wang, Z., You, F., Li, R., and Zhao, Z. Frieren: Efficient video-to-audio generation network with rectified flow matching. In The Thirty-eighth Annual Conference on Neural Information Processing Systems.

Xie, Z., Yu, S., He, Q., and Li, M. Sonicvisionlm: Playing sound with vision language models. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 26866–26875, 2024.

Yang, L., Qi, L., Feng, L., Zhang, W., and Shi, Y. Revisiting weak-to-strong consistency in semi-supervised semantic segmentation. In CVPR, pp. 7236–7246. IEEE, 2023.

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.

Yariv, G., Gat, I., Benaim, S., Wolf, L., Schwartz, I., and Adi, Y. Diverse and aligned audio-to-video generation via text-to-video model adaptation. In Proceedings of the AAAI Conference on Artificial Intelligence, volume 38, pp. 6639–6647, 2024.

Yu, L., Cheng, Y., Sohn, K., Lezama, J., Zhang, H., Chang, H., Hauptmann, A. G., Yang, M., Hao, Y., Essa, I., and Jiang, L. MAGVIT: masked generative video transformer. In CVPR, pp. 10459–10469. IEEE, 2023.

Yu, L., Lezama, J., Gundavarapu, N. B., Versari, L., Sohn, K., Minnen, D., Cheng, Y., Gupta, A., Gu, X., Hauptmann, A. G., Gong, B., Yang, M., Essa, I., Ross, D. A., and Jiang, L. Language model beats diffusion - tokenizer is key to visual generation. In ICLR. OpenReview.net, 2024.

Zhang, D., Zhang, H., Tang, J., Wang, M., Hua, X., and Sun, Q. Feature pyramid transformer. In ECCV (28), volume 12373 of Lecture Notes in Computer Science, pp. 323–339. Springer, 2020.

Zhang, X., Zhou, T., Zhang, X., Wei, J., and Tang, Y. Multi-scale diffusion: Enhancing spatial layout in high-resolution panoramic image generation. arXiv preprint arXiv:2410.18830, 2024a.

Zhang, Y., Gu, Y., Zeng, Y., Xing, Z., Wang, Y., Wu, Z., and Chen, K. Foleycrafter: Bring silent videos to life with lifelike and synchronized sounds. arXiv preprint arXiv:2407.01494, 2024b.

Zhao, Y., Gu, A., Varma, R., Luo, L., Huang, C.-C., Xu, M., Wright, L., Shojanazeri, H., Ott, M., Shleifer, S., et al. Pytorch fsdp: experiences on scaling fully sharded data parallel. arXiv preprint arXiv:2304.11277, 2023.

Zheng, H., Wang, Z., Yuan, J., Ning, G., He, P., You, Q., Yang, H., and Zhou, M. Learning stackable and skippable LEGO bricks for efficient, reconfigurable, and variable-resolution diffusion modeling. In ICLR. OpenReview.net, 2024a.

Zheng, Z., Peng, X., Yang, T., Shen, C., Li, S., Liu, H., Zhou, Y., Li, T., and You, Y. Open-sora: Democratizing efficient video production for all, March 2024b. URL https://github.com/hpcaitech/Open-Sora.

Zhou, Y., Wang, Q., Cai, Y., and Yang, H. Allegro: Open the black box of commercial-level video generation model. 2024. URL https://api.semanticscholar.org/CorpusID:273501978.

Zhuo, L., Du, R., Xiao, H., Li, Y., Liu, D., Huang, R., Liu, W., Zhao, L., Wang, F.-Y., Ma, Z., et al. Lumina-next: Making lumina-t2x stronger and faster with next-dit. arXiv preprint arXiv:2406.18583, 2024.

A. Basic Compositions of Lumina-Video

A.1. Loss function

Lumina-Video is trained with flow matching (Lipman et al., 2023; 2024; Liu et al., 2023b), a generative framework which builds a probability path from random noise $\mathbf{x}_0 \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ to the target data distribution $\mathbf{x}_1 \sim p_{\text{data}}$. The model is tasked with predicting the velocity field of samples $\mathbf{u}_t(\cdot)$ which can be used to reconstruct the data sample by solving the following Ordinary Differential Equation (ODE):

$$\frac{\mathrm{d}}{\mathrm{d}t}\mathbf{x}_t = \mathbf{u}_t(\mathbf{x}_t) \tag{3}$$

One commonly adopted form of the probability path is the linear path, which assumes $\mathbf{x}_t$ is a linear interpolation of noise and data sample:

$$\mathbf{x}_t = t\mathbf{x}_1 + (1-t)\mathbf{x}_0 \sim \mathcal{N}(t\mathbf{x}_1, (1-t)^2\mathbf{I}) \tag{4}$$

This assumption leads to a velocity field in the form of

$$\mathbf{u}_t(\mathbf{x}|\mathbf{x}_1) = \frac{\mathbf{x}_1 - \mathbf{x}}{1-t} \tag{5}$$

Equipped with Equation 4 and 5, the flow matching loss can be formulated as

$$\mathcal{L}(\theta) = \mathbb{E}_{t,\mathbf{x}_t,\mathbf{x}_1}\|\mathbf{u}_t^{\theta}(\mathbf{x_t}, t) - \mathbf{u}_t(\mathbf{x}_t|\mathbf{x}_1)\|^2 \tag{6}$$

A.2. Architecture

VAE. We adopt the 3D causal VAE of CogVideoX (Yang et al., 2024) for encoding and decoding videos between the pixel space and the latent space. Compared to the 2-D VAE used in Lumina-Next with spatial compression only, this VAE achieves higher efficiency by applying compression in the temporal dimension with less information leak from redundant frames.

Text Encoder. Following Lumina-Next, we utilize the Gemma-2-2B model (Team et al., 2024) as our text encoder. Despite being a lightweight text encoder, Gemma-2-2B excels at extracting visual semantics from natural language, enabling accurate and efficient alignment between textual descriptions and video content.

Multi-scale Next-DiT. The backbone of Lumina-Video is an improved version of Next-DiT (Zhuo et al., 2024), a flow-based diffusion transformer that incorporates the following key modifications to diffusion transformers: 1) Replacing 1D RoPE with 3D RoPE to instill more accurate positional prior in visual modeling; 2) Introducing sandwich normalization to control the magnitude of activations and stabilize the training process; 3) Incorporating Grouped-Query Attention to reduce computational demand; To adapt Next-DiT to video generation, Lumina-Video introduces Multi-scale Next-DiT, a transformative extension of Next-DiT to a multi-scale architecture, which is elaborated in Sec. 3.1.

B. Training Details

The AdamW optimizer (Loshchilov & Hutter, 2017), with weight decay set to 0.0 and betas $(0.9, 0.95)$, is employed. Furthermore, PyTorch FSDP (Zhao et al., 2023) with gradient checkpointing is utilized for reduced memory cost. To enhance training throughput, all video data are pre-encoded using a VAE encoder before training. The data are then clustered based on duration, ensuring that each global batch consists of samples with similar lengths.

C. Detailed Results

C.1. Full VBench Results

Table 4. Detailed results on VBench.

Model	Subject Consistency	Background Consistency	Temporal Flickering	Motion Smoothness	Dynamic Degree	Aesthetic Quality	Imaging Quality	Object Class
Proprietary Models								
Pika-1.0	96.94	97.36	99.74	99.50	47.50	62.04	61.87	88.72
Kling	98.33	97.60	99.30	99.40	46.94	61.21	65.62	87.24
Vidu	94.63	96.55	99.08	97.71	82.64	60.87	63.32	88.43
Gen-3 Alpha	97.10	96.62	98.61	99.23	60.14	63.34	66.82	87.81
Luma	97.33	97.43	98.64	99.35	44.26	65.51	66.55	94.95
Sora	96.23	96.35	98.87	98.74	79.91	63.46	68.28	93.93
Open-Source Models								
OpenSora Plan V1.3	97.79	97.24	99.20	99.05	30.28	60.42	56.21	85.56
OpenSora V1.2	96.75	97.61	99.53	98.50	42.39	56.85	63.34	82.22
VideoCrafter 2.0	96.85	98.22	98.41	97.73	42.50	63.13	67.22	92.55
Allegro	96.33	96.74	99.00	98.82	55.00	63.74	63.60	87.51
CogVideoX	96.23	96.52	98.66	96.92	70.97	61.98	62.90	85.23
Pyramid Flow	96.95	98.06	99.49	99.12	64.63	63.26	65.01	86.67
CogVideoX 1.5	96.87	97.35	98.88	98.31	50.93	62.79	65.02	87.47
Vchitect 2.0	96.83	96.66	98.57	98.98	63.89	60.41	65.35	86.61
HunyuanVideo	97.37	97.76	99.44	98.99	70.83	60.36	67.56	86.10
Lumina-Video-ss	96.06	97.26	98.63	98.90	67.13	62.27	64.58	91.03
Lumina-Video-ms	95.95	96.99	98.59	98.92	71.76	62.25	63.85	90.69

Model	Multiple Objects	Human Action	Color	Spatial Relationship	Scene	Appearance Style	Temporal Style	Overall Consistency
Proprietary Models								
Pika-1.0	43.08	86.20	90.57	61.03	49.83	22.26	24.22	25.94
Kling	68.05	93.40	89.90	73.03	50.86	19.62	24.17	26.42
Vidu	61.68	97.40	83.24	66.18	46.07	21.54	23.79	26.47
Gen-3 Alpha	53.64	96.40	80.90	65.09	54.57	24.31	24.71	26.69
Luma	82.63	96.40	92.33	83.67	58.98	24.66	26.29	28.13
Sora	70.85	98.20	80.11	74.29	56.95	24.76	25.01	26.26
Open-Source Models								
OpenSora Plan V1.3	43.58	86.80	79.30	51.61	36.73	20.03	22.47	24.47
OpenSora V1.2	51.83	91.20	90.08	68.56	42.44	23.95	24.54	26.85
VideoCrafter 2.0	40.66	95.00	92.92	35.86	55.29	25.13	25.84	28.23
Allegro	59.92	91.40	82.77	67.15	46.72	20.53	24.23	26.36
CogVideoX	62.11	99.40	82.81	66.35	53.20	24.91	25.38	27.59
Pyramid Flow	50.71	85.60	82.87	59.53	43.20	20.91	23.09	26.23
CogVideoX 1.5	69.65	97.20	87.55	80.25	52.91	24.89	25.19	27.30
Vchitect 2.0	68.84	97.20	87.04	57.55	56.57	23.73	25.01	27.57
HunyuanVideo	68.55	94.40	91.60	68.68	53.88	19.80	23.89	26.44
Lumina-Video-ss	68.32	97.67	90.16	67.27	56.08	23.64	25.66	28.22
Lumina-Video-ms	65.27	97.33	89.58	63.68	57.10	25.47	23.42	28.23

C.2. Full version of Figure 3

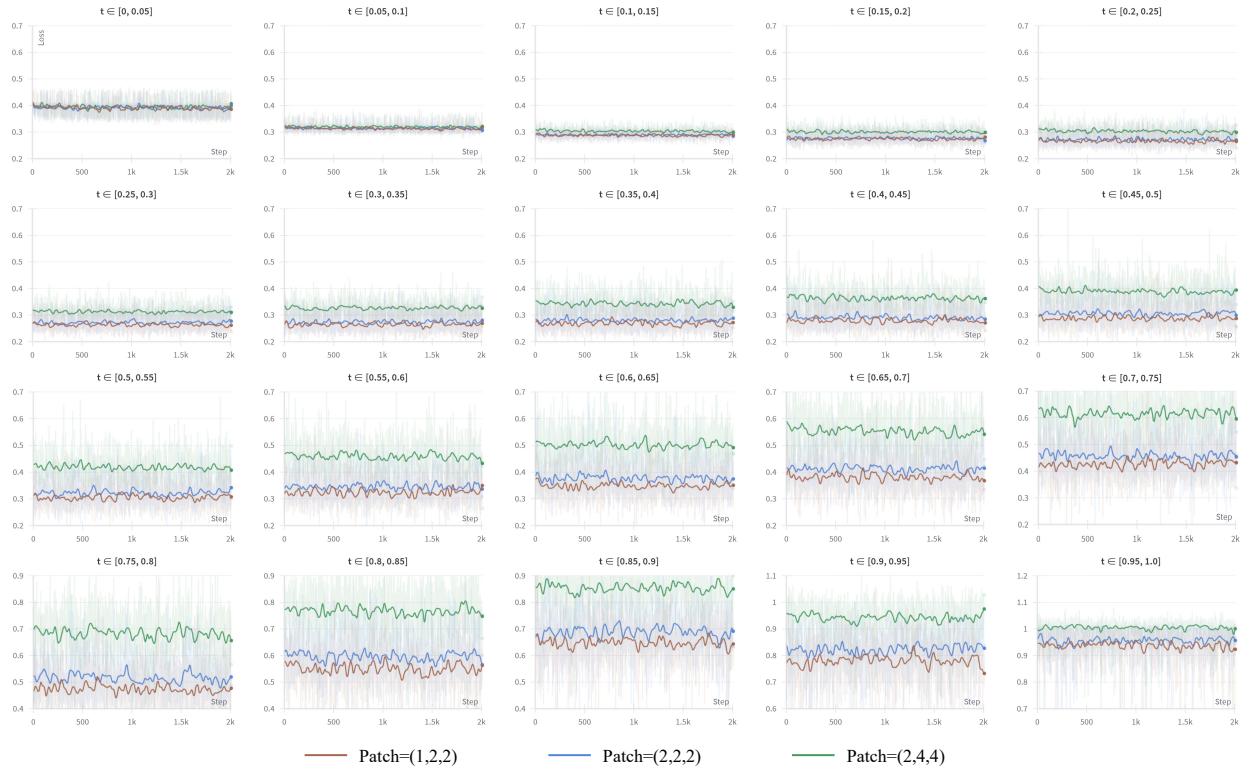

Figure 7. Complete figure of loss curves for different patch sizes at different denoising timesteps.

D. Video-to-Audio

In this section, we extend Lumina-Video with video-to-audio ability by designing Lumina-V2A to generate ambient sounds for silent video by synchronizing with visible scenes.

D.1. Background

Some existing works adopt a two-stage process to first align the video features with acoustic (Luo et al., 2024; Wang et al.) by unsupervised pretraining, and then introduce diffusion or flow-matching models to generate audio. Other approaches are proposed to first extract visual language (Wang et al., 2024) or time-varying features (Zhang et al., 2024b; Jeong et al., 2024; Lee et al., 2024; Xie et al., 2024) such as timestamps or energy curves from videos and then leverage pre-trained text-to-audio generation models to produce corresponding audio via trainable introduced adapters. Recent V2A works have achieved audio generation conditioned on both video and text (Polyak et al., 2024b; Cheng et al., 2024; Chen et al., 2024e), creating high-fidelity sound effects aligned with visual content. Our proposed V2A model aims to generate audios that are temporally synchronized with videos and semantically aligned with both video and text.

D.2. Model Architecture

As depicted in Figure 8, our Lumina-V2A model receives the video and text conditions to generate audio based on Next-DiT and rectified flow matching (Liu et al., 2023b). Specifically, the input audio waveform is first transformed into the 2D mel-spectrogram with an STFT operator and is then encoded by a pre-trained audio VAE encoder (Liu et al., 2023a) to obtain the compressed audio latents. Meanwhile, the pre-trained CLIP visual encoder and CLIP textual encoder (Fang et al., 2023; Radford et al., 2021) are employed to successively encode video and text into frame-level visual features and text embeddings. Next, visual features, text embeddings and audio latents are independently projected into modality-specific inputs to undergo the following Next-DiT blocks.

To efficiently integrate video, text, and audio modalities, we adopt a sequence of video-text-audio Next-DiT blocks to

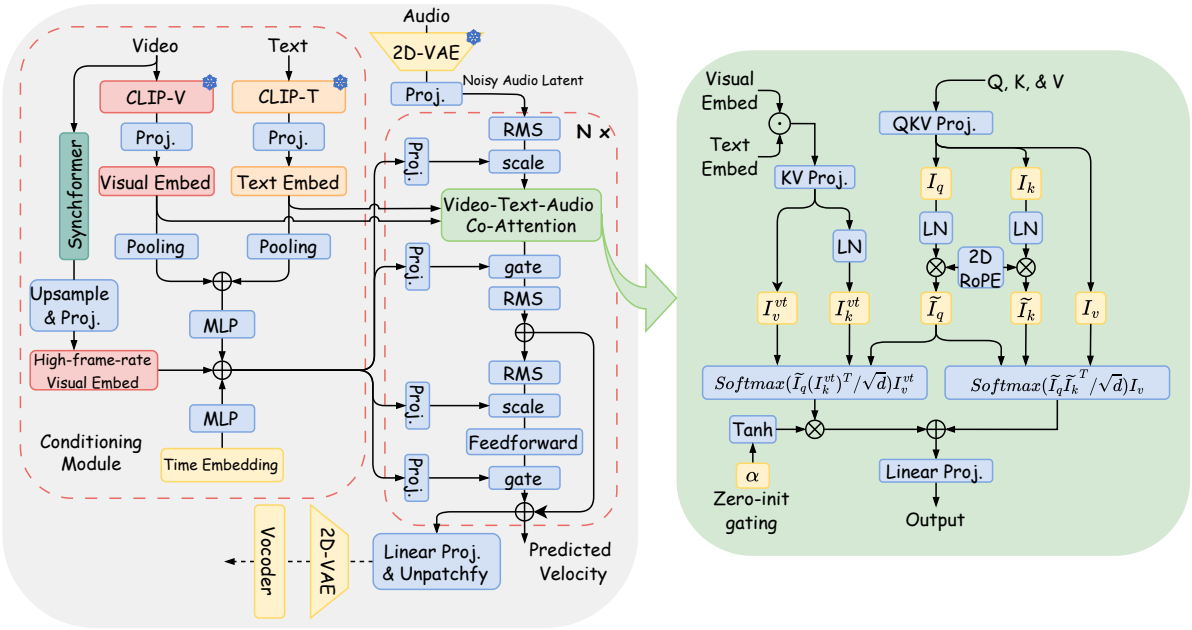

Figure 8. Illustration of Lumina-V2A Model based on Next-DiT

process the concatenated multimodal features via an inner co-attention mechanism. Within the co-attention module, the visual and text embeddings are concatenated together to interact with audio latent tokens like Lumina-Video. Additionally, It is important to ensure semantic alignment between audio and video-text and temporal synchronization between audio and video. To do so, we propose a multimodal conditioning module to integrate the time embedding, global visual and textual features, and high-frame-rate visual features from Synchformer (Iashin et al., 2024), forming a multimodal condition to be injected into Next-DiT blocks via scaling and gating operation.

During the inference stage, once the audio latent representation is generated by the proposed diffusion transformer, the pre-trained VAE decoder is used to reconstruct generated audio latents back to the mel-spectrogram that is then transformed into audio waveform via a pre-trained HiFi-GAN vocoder (Kong et al., 2020).

D.3. Training Data

We conduct experiments of our video-to-audio model on the VGGSound (Chen et al., 2020), a large-scale audio-visual dataset including 500 hours of videos with audio tracks in the wild and 310 classes. After filtering out invalid video IDs, we split the remaining dataset into around 180k videos for training, 2k videos for validation, and 15k for testing. To further improve the quality of VGGSound, we adopt the AV-Align score (Yariv et al., 2024) as the temporal alignment metric to select more aligned audio-visual pairs by setting the threshold as 0.2, resulting in about 110k high-quality video-audio pairs for future fine-tuning. By following the existing work (Cheng et al., 2024), we truncate each video clip to 8s duration during the training stage.