Mind the Time: Temporally-Controlled Multi-Event Video Generation

Ziyi Wu[1,2,3], Aliaksandr Siarohin[1], Willi Menapace[1], Ivan Skorokhodov[1],
Yuwei Fang[1], Varnith Chordia[1], Igor Gilitschenski[2,3,*], Sergey Tulyakov[1,*]
[1]Snap Research, [2]University of Toronto, [3]Vector Institute

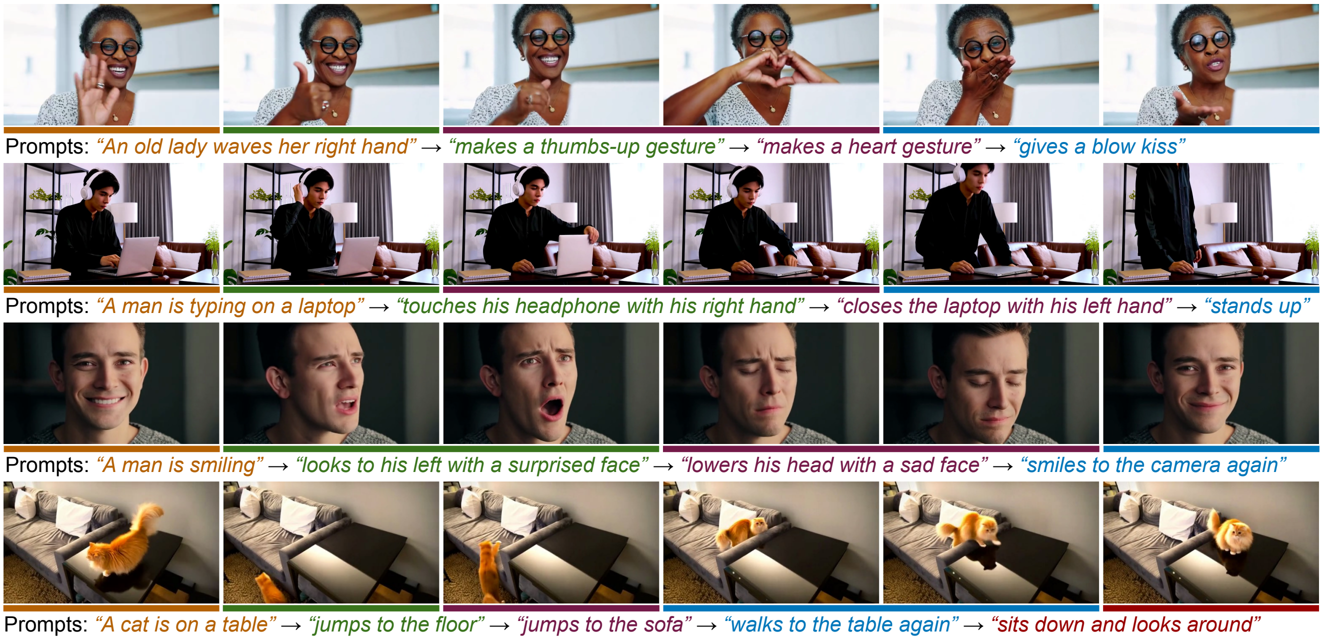

Prompts: "An old lady waves her right hand" → "makes a thumbs-up gesture" → "makes a heart gesture" → "gives a blow kiss"
Prompts: "A man is typing on a laptop" → "touches his headphone with his right hand" → "closes the laptop with his left hand" → "stands up"
Prompts: "A man is smiling" → "looks to his left with a surprised face" → "lowers his head with a sad face" → "smiles to the camera again"
Prompts: "A cat is on a table" → "jumps to the floor" → "jumps to the sofa" → "walks to the table again" → "sits down and looks around"

Figure 1. **Time-controlled multi-event video generation with MinT**. Given a sequence of event text prompts and their desired start and end timestamps, MinT synthesizes smoothly connected events with consistent subjects and backgrounds. In addition, it can control the time span of each event flexibly. Here, we show the results of sequential gestures, daily activities, facial expressions, and cat movements.

Abstract

Real-world videos consist of sequences of events. Generating such sequences with precise temporal control is infeasible with existing video generators that rely on a single paragraph of text as input. When tasked with generating multiple events described using a single prompt, such methods often ignore some of the events or fail to arrange them in the correct order. To address this limitation, we present MinT, a multi-event video generator with temporal control. Our key insight is to bind each event to a specific period in the generated video, which allows the model to focus on one event at a time. To enable time-aware interactions between event captions and video tokens, we design a time-based positional encoding method, dubbed ReRoPE. This encoding helps to guide the cross-attention operation. By fine-tuning a pre-trained video diffusion transformer on temporally grounded data, our approach produces coherent videos with smoothly connected events. For the first time in the literature, our model offers control over the timing of events in generated videos. Extensive experiments demonstrate that MinT outperforms existing open-source models by a large margin. Additional results and details are available at our project page.

1. Introduction

Recent research in video diffusion models [37] has achieved tremendous progress [8, 9, 13, 14, 26, 36, 69, 96]. These approaches typically rely on a single text prompt, and generate videos capturing only a single event. In contrast, real-world videos often comprise sequences of events with rich dynamics. Thus, achieving realism requires the ability to generate multiple events with fine-grained temporal control [63, 88].

A naive solution to multi-event video generation is to concatenate all event descriptions into a single, extended prompt, such as "A man raises his arms, lowers them down, and then moves them left and right". However, Fig. 2 shows that even state-of-the-art video models struggle to produce satisfactory results from such prompts. Some recent works tackle this problem in an autoregressive way [63, 87]. They generate each event individually with its own prompt, and

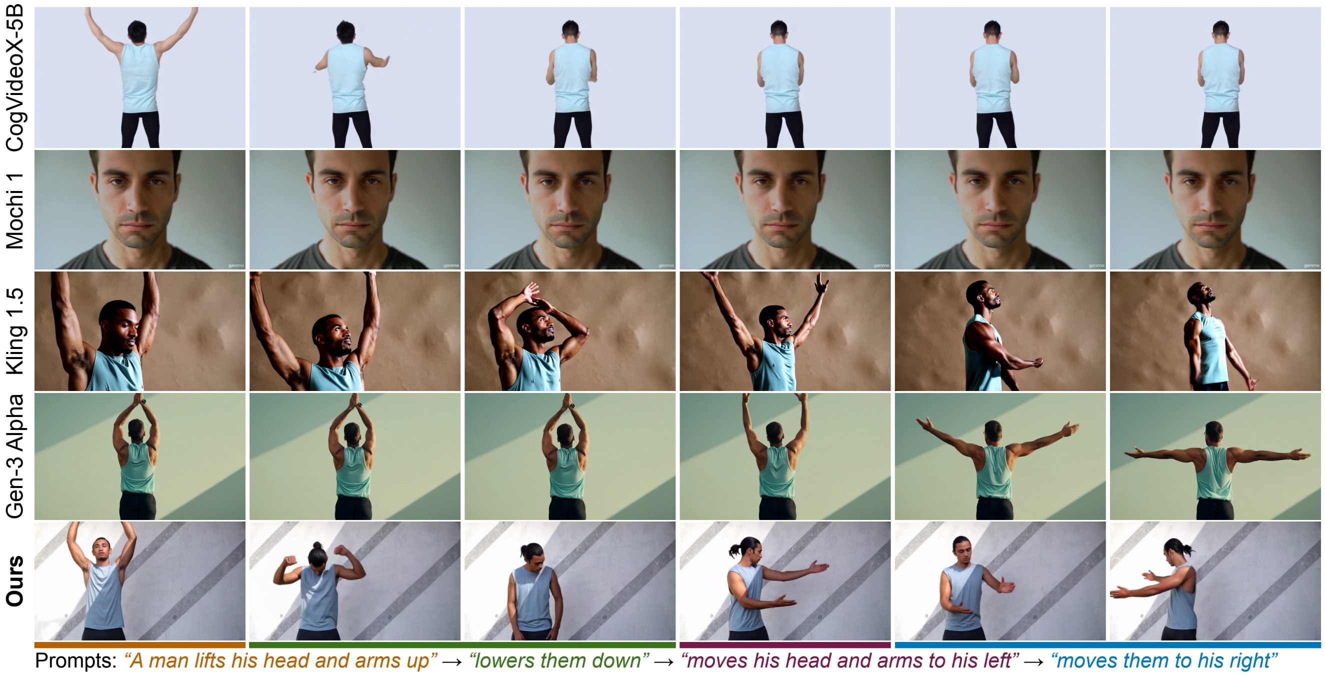

CogVideoX-5B
Mochi 1
Kling 1.5
Gen-3 Alpha
Ours
Prompts: "A man lifts his head and arms up" → "lowers them down" → "moves his head and arms to his left" → "moves them to his right"

Figure 2. **Multi-event video generation results from state-of-the-art video generators and MinT.** We run two open-source models CogVideoX-5B [100] and Mochi 1 [82], and two commercial models Kling 1.5 [2] and Gen-3 Alpha [1] with a text prompt containing four consecutive events. All of them only generate a subset of events while ignoring the remaining ones. In contrast, MinT generates a natural video with all events smoothly connected. Please refer to Appendix C.6 and our project page for more comparisons.

condition the model on the last frame of the previous event to ensure consistency. Yet, they often generate stagnated video frames with limited motion [22, 31]. Another line of work leverages personalized video generation to synthesize multiple event clips with consistent subjects [50, 54]. To get the final video they have to concatenate all the generated clips into one, leading to abrupt scene cuts. In addition, all existing methods present each event with a fixed-length video and cannot control the duration of individual events.

Recent work [47, 49] has shown that text-guided models often struggle with intricate spatial prompts, which can be improved by binding objects to spatial inputs (e.g., bounding boxes). Similarly, we hypothesize that the absence of explicit *temporal binding* precludes successful multi-event video generation in current models. Given a multi-event text prompt without timestamps, the generator must plan the time range of each event to form a video, which involves complicated reasoning. Inspired by the content-motion decomposition paradigm in video generation [83, 86], we propose to use **(i)** a global caption depicting content such as background and subject appearances, and **(ii)** a sequence of *temporal captions* [44] describing dynamic events, as our model input. Each temporal caption consists of a text description and the *start* and *end time* of the event. By providing temporally localized captions, the model can focus on one event at a time. In addition, our model processes all text prompts to generate a video in one shot, which ensures consistent subjects and smooth transitions between events.

Our resulting method, named **Min**d the **T**ime (MinT), is a temporally-grounded video generator built upon a pre-trained latent Diffusion Transformer (DiT) [66]. In each DiT block, we employ two cross-attention layers for global and temporal captions, respectively. To condition the model on a sequence of events, we concatenate the text embeddings of all temporal captions and perform cross-attention. The key challenge here is how to use the event timestamps to associate each caption with corresponding video tokens. Inspired by Rotary Position Embedding (RoPE) [81], we introduce Rescaled RoPE (ReRoPE) to guide the event caption to focus on attending to frames within its time range while ensuring a smooth transition between adjacent events.

In summary, this work makes four main contributions: **(i)** MinT, the first video generator that supports sequential event generation with time control. **(ii)** A novel training strategy that conditions the model on scene cuts, facilitating training on long videos and shot transition control. **(iii)** State-of-the-art multi-event video generation results in both text-only and image-conditioned settings on a hold-out set of our dataset and StoryBench [12]. **(iv)** An LLM-based prompt enhancer that extends short prompts to detailed global and temporal captions, from which we can generate videos with richer motion evaluated by VBench [39].

2. Related Work

Text-to-video diffusion models. With recent progress in diffusion models [35, 51, 78], text-to-video generation has achieved tremendous progress [8, 36, 37]. Earlier works inflate pre-trained image diffusion models by inserting temporal attention layers [7, 9, 13, 14, 26, 29, 42, 77, 89, 93, 95, 104]. They typically adopt a U-Net [75] model as the denoising network and run the diffusion process in a compressed latent space produced by a Variational Autoencoder (VAE) [43, 74, 101]. Recently, Transformer-based architecture [66, 85] has drawn increasing attention as it demonstrates better scalability in generating high-resolution and complex videos [11, 17, 27, 57, 61, 69, 76, 100]. Nevertheless, we identify the inability to generate sequential events as a common failure case in these models. By binding event captions to time and fine-tuning on temporally-grounded data, MinT greatly improves multi-event synthesis.

Story visualization. Traditionally, the goal of story visualization is to generate a sequence of images with consistent entities given multiple text prompts [58–60, 64, 73, 107]. Some recent works enhance the task by generating a video for each text prompt [24, 30, 48, 102, 106]. They usually leverage LLMs to plan the temporal ordering of events, and then run video personalization methods to generate clips with consistent character identities. However, these methods simply concatenate all generated clips to form a story, resulting in abrupt scene cuts between events [50, 54]. In this work, we tackle a different task that aims to generate videos of multiple events with natural transitions.

Multi-event video generation. Several studies have explored the generation of temporally consistent videos from multiple text prompts [22, 25, 31, 88]. The pioneering work, Phenaki [87], trains a masked Transformer to generate each event conditioning on its text prompt and frames from the preceding event. However, the autoregressive generation paradigm inevitably leads to quality degradation over longer sequences. FreeNoise [70] and MEVG [63] instead use previously generated clips to initialize the noise latent of the current clip, serving as a soft guidance for the model. A fundamental limitation of sequential generation approaches is that they generate all events with a fixed length [31]. In addition, their model lacks information about future events when generating the current event, preventing it from planning the entire video. On the contrary, MinT processes text prompts of all events together, allowing fine-grained control of event durations and generating globally coherent videos.

Rich captions for video generation. Previous large-scale video-text datasets usually comprise videos with short captions [6, 18]. Recent studies have shown that detailed captions are crucial for high-quality video generation [11, 15, 41, 100]. Yet, these datasets mainly focus on the appearance and spatial layout of all entities in a video. Closer to our task is the LVD-2M dataset [97], which labels sequential events in motion-rich videos. However, they only use text to describe the order of events, without localizing them in time. In this work, we are the first to enhance captions with precise timestamps for video generation. In addition, we study a previously overlooked scene cut annotation of video data, which further enhances the controllability of our model.

3. Method

Task formulation. Given a sequence of N^{e} temporally localized text prompts, $\{(c_n, t_n^{\text{start}}, t_n^{\text{end}})\}_{n=1}^{N^{\text{e}}}$, and N^{cut} cut timestamps, $\{t_n^{\text{cut}}\}_{n=1}^{N^{\text{cut}}}$, our goal is to generate a video containing all events following their text prompt c_n at the desired time range $(t_n^{\text{start}}, t_n^{\text{end}})$. The video is assumed to have no shot transition except at the input cut timestamps.

Overview. We build upon a pre-trained text-to-video Diffusion Transformer (DiT) [66] (Sec. 3.1). Our method, MinT, incorporates a temporally-aware cross-attention layer to enable event timestamp control (Sec. 3.2) and conditioning on video scene cuts (Sec. 3.3). Finally, we design a prompt enhancer that allows users to generate multi-event videos from simple prompts with our model (Sec. 3.4).

3.1. Background: Text-to-Video Latent DiT

Given a video, our latent DiT [66] first encodes it to video tokens $\boldsymbol{z}$ with a tokenizer [43]. Then, it adds Gaussian noise $\boldsymbol{\epsilon}_t$ to $\boldsymbol{z}$ to obtain a noisy sample $\boldsymbol{z}_t$, and trains a denoising network following the rectified flow formulation [51, 52]:

$$\mathcal{L}_{\text{DiT}} = ||\boldsymbol{v}_t - u_\theta(\boldsymbol{z}_t, t, \boldsymbol{y})||^2, \text{ where } \boldsymbol{v}_t = \boldsymbol{\epsilon}_t - \boldsymbol{z}. \quad (1)$$

Here, u_θ is implemented as a Transformer model [85] consisting of a stack of DiT blocks, and $\boldsymbol{y}$ denotes the conditioning signals such as text embeddings of a video caption. Similar to recent works [27, 69, 103], each DiT block in our base model contains a self-attention layer over video tokens, a cross-attention layer fusing video and text, and an MLP.

Rotary Position Embedding (RoPE). To indicate the position of video tokens in attention, our base model utilizes RoPE [81] due to its wide application in recent works [23, 46, 56, 100]. At a high level, given a sequence of N vectors $\{\boldsymbol{x}_n\}_{n=1}^{N}$, RoPE computes an angle θ_n for each vector $\boldsymbol{x}_n$ using its position n, and rotates $\boldsymbol{x}_n$ with θ_n to obtain $\tilde{\boldsymbol{x}}_n$:

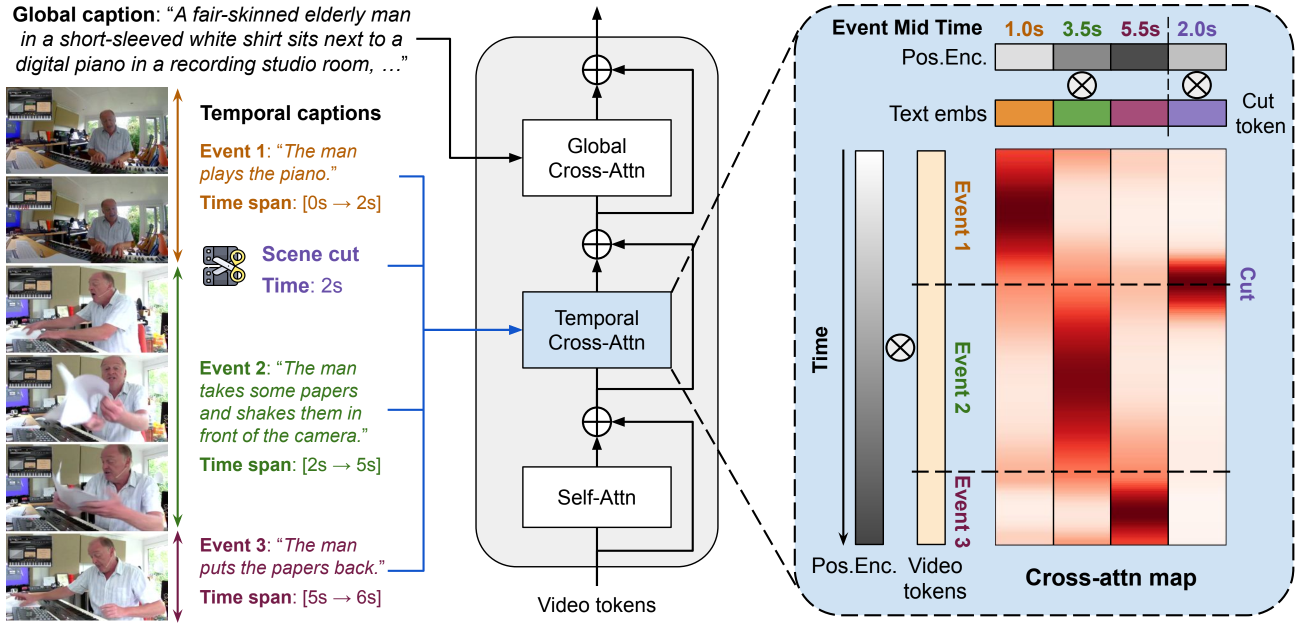
$$\theta_n = n\theta_{\text{base}}, \ \ \tilde{\boldsymbol{x}}_n = \text{RoPE}(\boldsymbol{x}_n, n) = \boldsymbol{x}_n e^{i\theta_n}, \quad (2)$$

where θ_{base} is a pre-defined base angle.[1] With RoPE, a vector $\boldsymbol{x}_n$ has a similar rotation angle with a vector $\boldsymbol{x}_m$ when n and m are close. Consequently, RoPE encourages nearby vectors to have a higher self-attention weight $A_{n,m}$:

$$\begin{aligned} A_{n,m} &= \text{Re}[\langle \tilde{\boldsymbol{x}}_n, \tilde{\boldsymbol{x}}_m \rangle] = \text{Re}[\langle \boldsymbol{x}_n, \boldsymbol{x}_m \rangle e^{i(n-m)\theta_{\text{base}}}] \\ &= \text{Re}[\text{RoPE}(\langle \boldsymbol{x}_n, \boldsymbol{x}_m \rangle, n-m)], \end{aligned} \quad (3)$$

where $A_{n,m}$ decreases monotonically with $|n-m|$ when $(n-m)\theta_{\text{base}} \in [-\pi/2, \pi/2]$. This usually holds in our DiT since the video tokens $\boldsymbol{z}$ are of low resolution. Please refer to Appendix A.2 for a rigorous discussion. In our video DiT, RoPE is only applied in the self-attention. There is no positional encoding in the video-text cross-attention as the input text prompt is expected to describe the entire video.

[1] In fact, RoPE uses a list of angles $\boldsymbol{\theta} \in \mathbb{R}^{h/2}$ to rotate each element of a vector $\boldsymbol{x} \in \mathbb{R}^h$ separately. We treat it as a single angle for simplicity in this paper, as all dimensions of $\boldsymbol{\theta}$ changes monotonically with n [81].


Global caption: "A fair-skinned elderly man in a short-sleeved white shirt sits next to a digital piano in a recording studio room, …"
Temporal captions
Event 1: "The man plays the piano."
Time span: [0s → 2s]
Scene cut
Time: 2s
Event 2: "The man takes some papers and shakes them in front of the camera."
Time span: [2s → 5s]
Event 3: "The man puts the papers back."
Time span: [5s → 6s]
Global Cross-Attn
Temporal Cross-Attn
Self-Attn
Video tokens
Event Mid Time
1.0s
3.5s
5.5s
2.0s
Pos.Enc.
Text embs
Cut token
Time
Event 1
Event 2
Event 3
Cut
Pos.Enc.
Video tokens
Cross-attn map

(a) Time-based caption data
(b) Video DiT Block
(c) Temporal Cross-Attn Layer

Figure 3. **MinT framework.** (a) Our model takes in a global caption describing the overall video, and a list of temporal captions specifying the sequential events. We bind each event to a time range, enabling temporal control of the generated events. (b) To condition the video DiT on temporal captions, we introduce a new temporal cross-attention layer in each DiT block, which (c) concatenates the text embedding of all event prompts and leverages a time-aware positional encoding (Pos.Enc.) method to associate each event to its corresponding frames based on the event timestamps. MinT supports an additional scene cut conditioning, which can control the shot transition of the video.

3.2. Temporally Aware Video DiT

Existing text-guided video diffusion models only take in one global text prompt for a video. As shown in Fig. 3 (a), we further input a sequence of temporal captions that bind each event to an exact time range. The decomposition of global and temporal captions resembles the classic content-motion disentanglement in video generation [83, 86], providing a clearer guidance of video dynamics to the model.

Temporal cross-attention. To condition MinT on temporal captions, we add a new temporal cross-attention layer between the original self-attention and cross-attention layers as shown in Fig. 3 (b). Prior works [47, 90, 92] show that such design enables fast adaptation to new spatial conditioning input, and we show that it also works for temporal conditioning. We first extract text embeddings $\boldsymbol{e}^{\mathrm{c}}_n \in \mathbb{R}^{L^{\mathrm{c}} \times D^{\mathrm{c}}}$ for each event text prompt c_n, where L^{c} and D^{c} are the text length and the embedding dimension, respectively. Then, we apply positional encoding to each $\boldsymbol{e}^{\mathrm{c}}_n$ to indicate its time span $[t^{\mathrm{start}}_n, t^{\mathrm{end}}_n]$, and concatenate them along the sequence dimension to perform cross-attention with video tokens:

$$\begin{aligned}\tilde{\boldsymbol{e}}^{\mathrm{c}}_n &= \mathrm{Pos.Enc.}(\boldsymbol{e}^{\mathrm{c}}_n, t^{\mathrm{start}}_n, t^{\mathrm{end}}_n),\\ \tilde{\boldsymbol{z}} &= \mathrm{XAttn}(\boldsymbol{z}, \mathrm{Concat}([\tilde{\boldsymbol{e}}^{\mathrm{c}}_1, \tilde{\boldsymbol{e}}^{\mathrm{c}}_2, ..., \tilde{\boldsymbol{e}}^{\mathrm{c}}_{N^{\mathrm{e}}}])).\end{aligned} \tag{4}$$

Apart from positional encoding, an intuitive way to indicate event time range is hard masking, where we only allow $\boldsymbol{e}^{\mathrm{c}}_n$ to attend to video tokens within $[t^{\mathrm{start}}_n, t^{\mathrm{end}}_n]$. However, for frames close to an event transition point, it is beneficial to receive information from both events to synthesize a smooth transition. Therefore, we decide to use RoPE to serve as soft masking to guide the text embedding of each event.

Intuitively, we want the temporal cross-attention to have three key properties: **(i)** For video tokens within the time span of an event, they should always attend the most to the text embedding of this event. **(ii)** For an event, the attention weight should peak with the video token at the midpoint of its time span, and then decrease towards the boundary of the event. **(iii)** The video token at the transition point between two events should attend equally to their text embeddings, which helps the model localize the event boundary.

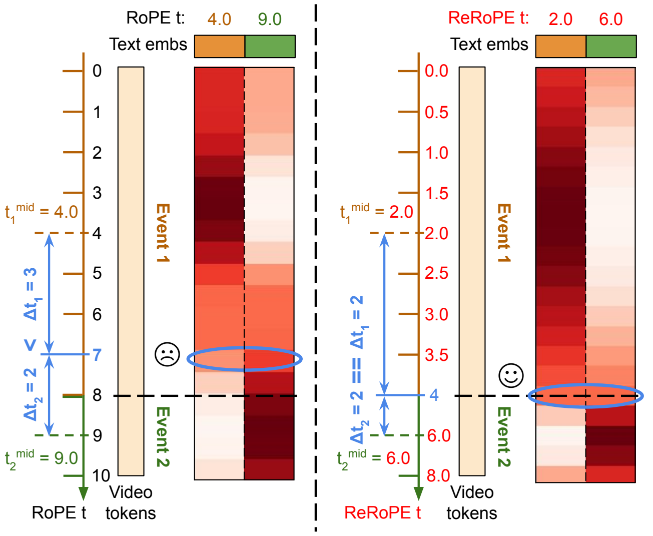
Below, we show that vanilla RoPE fails to achieve (i) and (iii), necessitating a new positional encoding for this task.

Vanilla temporal RoPE. We start from the standard RoPE in Eq. (2). For a video token $\boldsymbol{z}_{[t,\cdot,\cdot]}$ at any spatial location on frame t, we only use the timestamp t to determine its rotation angle θ since we focus on temporal correspondence here. For an event happening in $[t^{\mathrm{start}}_n, t^{\mathrm{end}}_n]$, a natural way to encode its text embedding is using its middle timestamp $t^{\mathrm{mid}}_n = (t^{\mathrm{start}}_n + t^{\mathrm{end}}_n)/2$. Therefore, the vanilla RoPE is as:

$$\tilde{\boldsymbol{z}}_{[t,\cdot,\cdot]} = \mathrm{RoPE}(\boldsymbol{z}_{[t,\cdot,\cdot]}, t), \ \ \tilde{\boldsymbol{e}}^{\mathrm{c}}_n = \mathrm{RoPE}(\boldsymbol{e}^{\mathrm{c}}_n, t^{\mathrm{mid}}_n), \tag{5}$$

$$\mathrm{Attn}(\tilde{\boldsymbol{z}}_{[t,\cdot,\cdot]}, \tilde{\boldsymbol{e}}^{\mathrm{c}}_n) = \mathrm{Re}[\mathrm{RoPE}(\langle \boldsymbol{z}_{[t,\cdot,\cdot]}, \boldsymbol{e}^{\mathrm{c}}_n \rangle, t - t^{\mathrm{mid}}_n)] \tag{6}$$

Such design satisfies property (ii), while violating the other properties as shown in Fig. 4 (a). In this example, frame 7 belonging to the first event is closer to t^{mid}_2 than t^{mid}_1 and thus has a higher attention weight with the second event. In addition, frame 8 which is at the intersection of two events



(a) Vanilla RoPE. Frame 7 has higher attention wight with **Event 2**.

(b) Our ReRoPE (rescaled to L=4). Same attention weight at event border.

Figure 4. **Comparison of vanilla RoPE and our Rescaled RoPE.** We use the same random vector for video tokens and text embeddings to only visualize the bias introduced by positional encoding. (a) Vanilla RoPE uses raw timestamps as the rotation angle, where frames within one event might be biased to the wrong text. (b) We instead rescale all events to have the same length L, so that video tokens always attend the most to the current event. In addition, frames at event boundaries attend to adjacent events equally.

attends to the second event more than the first one. As a result, the model cannot locate the correct event boundary.

Rescaled RoPE (ReRoPE). When adjacent events have different durations, their midpoints' distance to the event boundary also becomes different, causing vanilla RoPE to fail. Therefore, we propose to rescale all events to the same length L and recompute timestamps for encoding in Eq. (5). For a timestamp t lying in the n-th event, we transform it as:

$$\tilde{t} = \frac{(t - t_n^{\text{start}})L}{t_n^{\text{end}} - t_n^{\text{start}}} + (n-1)L, \quad \text{s.t. } t_n^{\text{start}} \le t \le t_n^{\text{end}}. \tag{7}$$

Using Eq. (7) for both video tokens and events, we have:

$$\tilde{t} - \tilde{t}_n^{\text{mid}} = \left(\frac{t - t_n^{\text{start}}}{t_n^{\text{end}} - t_n^{\text{start}}} - \frac{1}{2} \right) L. \tag{8}$$

As we show in Appendix A.2, our ReRoPE design achieves all three desired properties in the temporal cross-attention.

Inspired by Positional Interpolation [16], we set a fixed value for L, so that videos of different lengths are rescaled to the same length in Eq. (8). As a result, ReRoPE always induces the same attention bias to temporal cross-attention, making the layer invariant to the actual video length.

3.3. Scene Cut Conditioning

Prior large-scale video datasets usually remove videos with scene cuts or split them into shorter clips [18, 41, 94]. Indeed, training a generator on videos with cuts may lead to undesired scene transitions in generated videos.

Typically, professionally edited videos contain frequent cuts, and excluding them in training may lose valuable information. Removing such clips also reduces the amount of training data significantly (in our data 20% of clips contain cuts). But most importantly, it makes a model unable to use such a valuable cinematographic effect, leading to temporally cropped videos. Prior image generators face a similar issue with image cropping [68], where the model may learn to generate "cropped" images with out-of-frame objects. Based on these insights, we decide to keep all the videos, while explicitly conditioning the model on the timestamps of cuts. Once the model learns such conditioning, we can input zeros during inference to enforce a cut-free video.

We treat a scene cut as a special event with the same content and equal start and end timestamps. To condition MinT on it, we initialize a learnable vector $\boldsymbol{e}^{\text{cut}} \in \mathbb{R}^{1 \times D^c}$, apply ReRoPE with its timestamp t_n^{cut} transformed by Eq. (7), and concatenate it with the text embeddings of temporal captions to perform cross-attention with video tokens:

$$\tilde{\boldsymbol{e}}_n^{\text{cut}} = \text{ReRoPE}(\boldsymbol{e}_n^{\text{cut}}, t_n^{\text{cut}}, t_n^{\text{cut}}),$$
$$\tilde{\boldsymbol{z}} = \text{XAttn}(\boldsymbol{z}, \text{Concat}([\tilde{\boldsymbol{e}}_1^c, ..., \tilde{\boldsymbol{e}}_{N^e}^c, \tilde{\boldsymbol{e}}_1^{\text{cut}}, ..., \tilde{\boldsymbol{e}}_{N^{\text{cut}}}^{\text{cut}}])). \tag{9}$$

As we show in the ablation (Sec. 4.5), this design greatly reduces undesired scene transitions when they are not requested, and allows a practitioner to use them when needed.

3.4. Prompt Enhancer

MinT offers video generation with precise control of event timing. Yet, in certain applications starting from a single prompt can be more desirable. Prior works demonstrated that LLMs can generate physically meaningful spatial layout of scenes from text prompts [49, 50]. Similarly, we show that LLMs can plan the temporal structure of multi-event videos. Given a short text, we prompt LLMs to extend it to a detailed global caption and several event captions with their time span. Then, our model can generate a video with rich motion content from the enhanced prompts.

4. Experiments

Our experiments aim to answer the following questions: **(i)** Can MinT control event timing in both text-to-video (T2V) and image-to-video (I2V) settings? (Sec. 4.2 & Sec. 4.3) **(ii)** Does prompt enhancement lead to high-quality multi-event videos from a single prompt? (Sec. 4.4) **(iii)** What is the impact of each design choice in our framework? (Sec. 4.5)

4.1. Experimental Setup

We list some key aspects of our experimental setup here. For full details, please refer to Appendix B.

Training data. Existing video datasets with time-based captions usually come from dense video captioning [44, 105]. However, these datasets are limited in scale, which

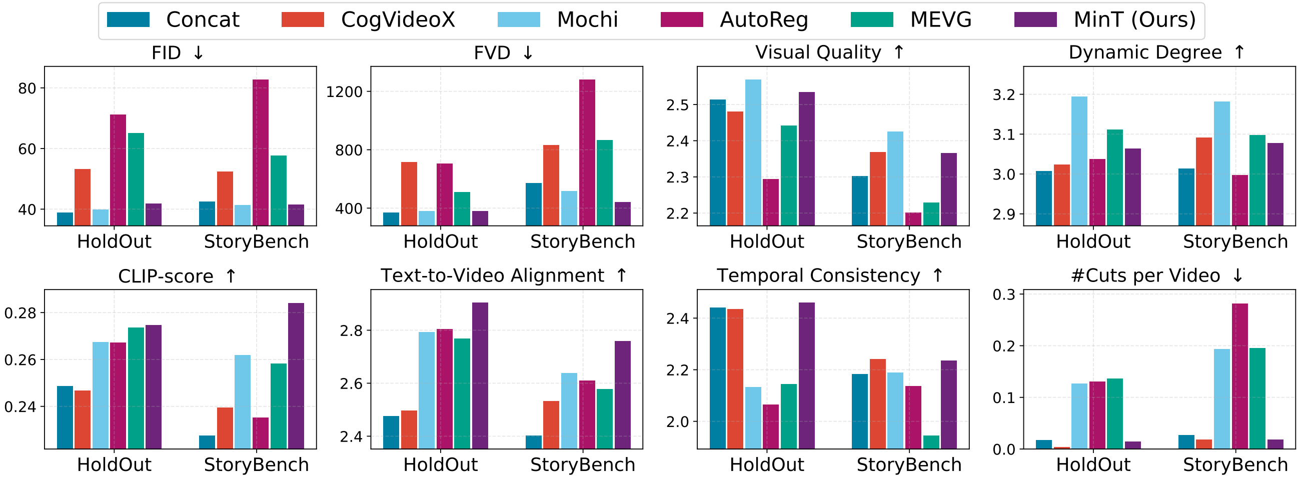


Figure 5. **T2V results on HoldOut and StoryBench.** For CogVideoX and Mochi we concatenated the events into a single prompt, similar to the Concat baseline. Metrics in the first row measure visual quality, while those in the second row focus on the text alignment and transition smoothness between events. MinT performs the best in event-related metrics while maintaining a high visual quality.

are impossible to fine-tune a large-scale video generator on. Therefore, we manually annotate temporal events on videos sourced from existing datasets [18, 98], resulting in around 200k videos, where we hold out 2k videos for evaluation. To condition the model on scene cuts, we run TransNetV2 [80] to detect scene boundaries on annotated videos.

Evaluation datasets. We leverage the 2k holdout videos as our primary benchmark (dubbed *HoldOut*). We also test on the *StoryBench* [12] dataset, which annotates temporal captions similar to ours. We filter out videos with only a single event, leading to around 3k testing samples. Finally, to test MinT's ability in generating motion-rich videos from short prompts, we utilize prompt lists from *VBench* [39].

Baselines. To show that current video models are not capable of generating multi-event videos, we design a straightforward method, called *Concat*, that simply concatenates all the prompts together. We apply it to both our base model and state-of-the-art open-source models CogVideoX [100] and Mochi [82]. We also compare to approaches with code available and are designed to generate smoothly connected events. *MEVG* [63] is the state-of-the-art multi-event video generation method. It generates each event from its prompt separately. To ensure smooth transitions, it runs DDIM inversion [79] on the previously generated event as the noise initialization for the current event. We also design a baseline that fine-tunes an image-conditioned video diffusion model to generate events autoregressively (dubbed *AutoReg*). To separate the impact of architecture from the method, we implement both MEVG and AutoReg on top of our base model ensuring a fair comparison. Notably, since no baselines can control event timing, we simply set all events to have the same length to make the comparison possible.

Evaluation metrics. We focus on three dimensions: visual quality, text alignment, and event transition smoothness. We report common metrics such as FID [33], FVD [84] for visual quality, and per-frame CLIP-score [32] for text alignment. In addition, we leverage a state-of-the-art video quality assessment model, VideoScore [28] as it has been shown to produce results consistent with human evaluators. We take the *visual quality* and *dynamic degree* output for visual quality, the *text-to-video alignment* output for text alignment, and the *temporal consistency* output for event transition smoothness. Notably, since we care about event generation, we compute text alignment between temporal captions and video clips cropped out based on the event span. Finally, we run TransNetV2 to detect the cuts in generated videos to measure event transition smoothness.

Implementation details. MinT builds upon a pre-trained latent video DiT similar to [46, 103]. It generates videos of 512×288 resolution and up to 12 seconds. We fine-tune the entire model with the AdamW optimizer [55] and a batch size of 512 for 12k steps. For inference, we run 256 denoising steps with a classifier-free guidance [34] scale of 8.

4.2. Text-to-Video Generation

Fig. 5 presents the quantitative results on HoldOut and StoryBench datasets. Fig. 6 shows a qualitative comparison. Compared to Concat which shares the same base model as ours, MinT achieves slightly lower visual quality on HoldOut and better results on StoryBench. This is because StoryBench prompts are out-of-distribution to our model. Despite this, time-based captions help MinT generate a video with a good temporal structure. On the other hand, we generate events with much higher text alignment. We draw similar observations when comparing MinT to the Concat baseline based on CogVideoX and Mochi. Overall, this proves that our model acquires the new capability of sequential event generation while maintaining high visual quality.

As for multi-event generation methods, AutoReg and MEVG greatly improve the text alignment, as they generate each event from its prompt separately. Yet, AutoReg has much lower visual quality, since conditioning on generated

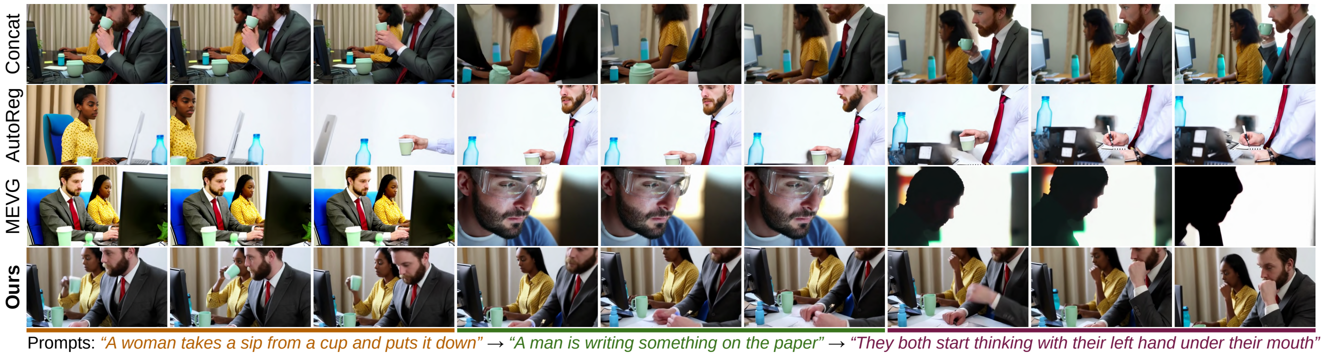

Concat
AutoReg
MEVG
Ours
Prompts: "A woman takes a sip from a cup and puts it down" → "A man is writing something on the paper" → "They both start thinking with their left hand under their mouth"

Figure 6. **Qualitative results of T2V.** Concatenating all events into a single prompt (Concat) can only generate the first event. Autoregressive generation (AutoReg) suffers from video stagnation and cannot generate the third event. MEVG struggles to preserve the person's identity and produces abrupt event transitions. MinT is the only method that generates all events with smooth transitions and consistent content. See Appendix C.1 for more qualitative results and our project page for video results.

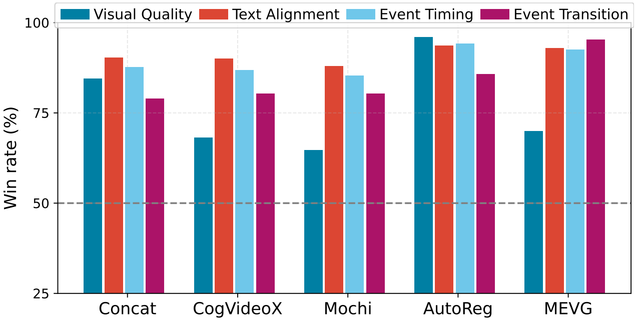


Figure 7. **Human preference evaluation against T2V baselines.** MinT is better or competitive in visual quality, and significantly outperforms baselines in the other three event-related metrics.

frames leads to artifacts such as video stagnation. MEVG resolves this issue with frame inversion. However, it often generates abrupt transitions between events as indicated by the large number of cuts. In fact, we found that the inversion technique in MEVG only works well when two consecutive event captions have a similar structure, e.g., the same subject doing different actions. When two captions have a subject change such as in Fig. 6, the generated events usually contain completely different characters. Overall, MinT achieves the best balance between video quality, event localization, and temporal smoothness of the video. See Appendix C.6 for comparisons with commercial models.

Human evaluation. We conduct a user study using 200 randomly sampled prompts from HoldOut. We ask the participants to express their preference when presented with paired samples from MinT and each baseline, gathering votes from 5 users per sample. Results in Fig. 7 show that MinT has better or competitive visual quality, while generating events with significantly higher text alignment, timing accuracy, and transition smoothness.

Event time control. MinT supports fine-grained control of event timing. Please refer to Appendix C.4 for our results.

4.3. Image-conditioned Video Generation

We evaluate the model's ability to animate entities in an existing image to perform sequential events. Following [12], models have access to the ground-truth initial frame of testing videos as well as the event text prompts.

Method	FID ↓	FVD ↓	VQ ↑	DD ↑	CLIP-T ↑	TA ↑	TC ↑	#Cuts ↓
Dataset: HoldOut								
MEVG	57.57	495.75	2.56	**3.39**	0.266	2.72	2.25	0.108
Ours	**22.04**	**218.21**	**2.60**	3.30	**0.272**	**3.00**	**2.47**	**0.025**
Dataset: StoryBench								
MEVG	56.51	732.94	3.27	**3.80**	0.265	2.83	3.03	0.150
Ours	**21.85**	**314.59**	**3.36**	3.76	**0.273**	**3.37**	**3.29**	**0.014**

Table 1. **I2V results on HoldOut and StoryBench.** VQ, DD, TA, and TC stand for visual quality, dynamic degree, text-to-video alignment, and temporal consistency from VideoScore. #Cuts is the average number of cuts per video. Similar to T2V, MinT also achieves better visual quality and smooth event transition.

Settings. The same datasets and metrics as in the T2V setting are employed. We compare with the best baseline, *MEVG*. It has an image-conditioned variant that replicates the initial frame to form a pseudo video. For MinT, we finetune it to condition on an image by concatenating the image with the noisy latent similar to prior works [8, 96].

Results. Tab. 1 presents the multi-event image animation results on HoldOut and StoryBench datasets. We draw similar observations as in the T2V setting. MinT achieves either better or competitive results in visual quality, while performing significantly better in text alignment with event captions and temporal smoothness of event transitions.

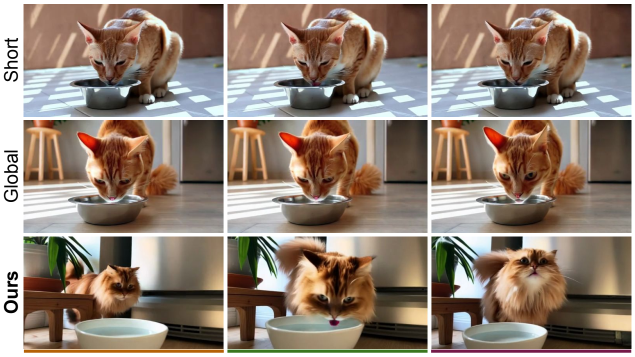
4.4. Prompt Enhanced Video Generation

MinT introduces a new dimension to prompt enhancement, where users can control the amount of motion in the generated video via temporal captions. We show that this process can be automated by an LLM. This enables users to generate more interesting videos from a short prompt.

Dataset. Since we are interested in the motion of generated videos, we take the list of prompts from the *Dynamic Degree* evaluation dimension on VBench [39]. These prompts are diverse and always contain subjects performing non-static actions. Yet, they are all short with around 10 words.

Method	Subject Consist. ↑	Background Consist. ↑	Aesthetic Quality ↑	Imaging Quality ↑	Motion Smooth ↑	Dynamic Degree ↑
Short	0.857	0.939	0.498	0.583	0.995	0.481
Global	0.890	0.950	0.541	0.613	0.995	0.517
Ours	0.900	0.950	0.544	0.609	0.988	**0.711**

Table 2. **Prompt enhancement results on VBench.** Consist. means consistency. The first four metrics measure video quality, while we focus on the motion of generated videos. MinT generates videos with significantly higher dynamics degree and competitive visual quality and motion smoothness.



Prompts: *"A cat walks towards a bowl"* → *"laps water with tongue"* → *"lifts its head"*

Figure 8. **Qualitative comparison of prompt enhancement results.** The original short prompt is "a cat drinking water".

Prompt enhancer. We prompt GPT-4 [3] to extend the short prompt to a detailed global caption and temporal captions. Please refer to Appendix C.2 for the prompt we use.

Baselines and evaluation metrics. We compare with videos generated by our base model using the original short prompts (dubbed *Short*). To disentangle the effect of global caption and temporal captions, we also compare to videos generated by our base model using the enhanced global caption (named *Global*). For evaluation, we compute six metrics from the official VBench test suite, which focus on visual quality, temporal smoothness, and motion richness.

Results. Tab. 2 demonstrates the video generation results from enhanced prompts on VBench. Global performs consistently better than Short, proving that using a detailed prompt is indeed beneficial. When equipped with additional temporal captions, MinT achieves competitive results with baselines in visual quality and motion smoothness, while scoring significantly higher in dynamic degree. Fig. 8 shows a qualitative example, where we turn a single-action prompt into a coherent video with three actions. Please refer to Appendix C.2 for more qualitative results.

4.5. Ablation Study

We study the effect of each component in our model in Tab. 3. All ablations are conducted on HoldOut.

Time conditioning. We examine different ways to condition the model on the event time span. *Concat time* runs an MLP to embed timestamps to high dimensional features, and then concatenates them with text embeddings of temporal captions. However, since our base model uses RoPE, the video tokens do not contain absolute positional information. Therefore, doing cross-attention with time-embedded text features cannot associate events to video frames, leading to significantly worse text alignment with event captions. *Hard attn mask* adopts hard masking in the temporal cross-attention, where events only attend to frames within its time range. This enables synthesizing events at desired time periods. However, hard masking prevents video tokens at event boundaries from attending to upcoming events, resulting in abrupt event transitions thus lower temporal consistency and more scene cuts. Finally, *Vanilla RoPE* encodes video tokens and text embeddings of events with raw timestamps. As discussed in Fig. 4, it fails to accurately locate event borders, which degrades the control of event timing as indicated by the lower text alignment scores.

Method	VQ ↑	DD ↑	CLIP-T ↑	TA ↑	TC ↑	#Cuts ↓
Full Model	2.56	3.32	**0.270**	**2.92**	2.44	0.026
Concat time	2.53	3.31	0.249	2.42	2.33	0.075
Hard attn mask	2.45	3.34	0.260	2.68	2.30	0.069
Vanilla RoPE	2.54	3.32	0.262	2.79	2.42	0.030
ReRoPE (L=4)	2.54	3.33	0.264	2.88	2.43	0.029
ReRoPE (L=16)	2.55	3.32	0.265	2.90	2.44	0.025
No cut condition	2.54	3.33	0.268	2.89	2.34	0.084

Table 3. **Ablation results on HoldOut.** We study different conditioning mechanisms for event time span, the rescale length L in ReRoPE, and the use of scene cut conditioning. VQ, DD, TA, and TC stand for visual quality, dynamic degree, text-to-video alignment, and temporal consistency from VideoScore. #Cuts is the average number of scene cuts per video.

ReRoPE rescaling length L. By default, we set $L = 8$. Tab. 3 shows that using $L = 4$ or 16 achieves similar results. This indicates that the model is insensitive to this hyper-parameter. Please refer to Appendix A.3 for more discussions on ReRoPE with different values of L.

Scene cut conditioning. In the last row of Tab. 3, we remove the scene cut conditioning during training. As discussed in Sec. 3.3, without access to scene cut information, the model will introduce undesired shot transitions in the generated video. Indeed, this variant has similar visual quality and text alignment as our full model, but scores much lower in temporal consistency and generates more cuts. Please refer to Appendix C.3 for more analysis.

5. Conclusion

We present MinT, a framework for multi-event video generation with event timing control. Our method employs a unique positional encoding method to guide the temporal dynamics of the video, resulting in smoothly connected events and consistent subjects. Equipped with LLMs, we further design a prompt enhancer that can generate motion-rich videos from a simple prompt. We view our work as an important step towards controllable content creation tools. We discuss our limitations and failure cases in Appendix D.

Acknowledgments

We would like to thank Tsai-Shien Chen, Alper Canberk, Yuanhao Ban, Sherwin Bahmani, Moayed Haji Ali, and Xijie Huang for valuable discussions and support.

References

[1] Gen-3 Alpha. https://runwayml.com/research/introducing-gen-3-alpha, 2024. Accessed: 2024-10-24. 2, 9, 10

[2] Kling1.5. https://klingai.com/, 2024. Accessed: 2024-10-24. 2, 9, 10

[3] Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. GPT-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023. 8, 6

[4] Lisa Anne Hendricks, Oliver Wang, Eli Shechtman, Josef Sivic, Trevor Darrell, and Bryan Russell. Localizing moments in video with natural language. In *ICCV*, 2017. 2

[5] Jimmy Lei Ba. Layer Normalization. *arXiv preprint arXiv:1607.06450*, 2016. 1

[6] Max Bain, Arsha Nagrani, Gül Varol, and Andrew Zisserman. Frozen in time: A joint video and image encoder for end-to-end retrieval. In *ICCV*, 2021. 3

[7] Omer Bar-Tal, Hila Chefer, Omer Tov, Charles Herrmann, Roni Paiss, Shiran Zada, Ariel Ephrat, Junhwa Hur, Yuanzhen Li, Tomer Michaeli, et al. Lumiere: A space-time diffusion model for video generation. *arXiv preprint arXiv:2401.12945*, 2024. 2

[8] Andreas Blattmann, Tim Dockhorn, Sumith Kulal, Daniel Mendelevitch, Maciej Kilian, Dominik Lorenz, Yam Levi, Zion English, Vikram Voleti, Adam Letts, et al. Stable video diffusion: Scaling latent video diffusion models to large datasets. *arXiv preprint arXiv:2311.15127*, 2023. 1, 2, 7

[9] Andreas Blattmann, Robin Rombach, Huan Ling, Tim Dockhorn, Seung Wook Kim, Sanja Fidler, and Karsten Kreis. Align your latents: High-resolution video synthesis with latent diffusion models. In *CVPR*, 2023. 1, 2

[10] Tim Brooks, Aleksander Holynski, and Alexei A Efros. Instructpix2pix: Learning to follow image editing instructions. In *CVPR*, 2023. 4

[11] Tim Brooks, Bill Peebles, Connor Holmes, Will DePue, Yufei Guo, Li Jing, David Schnurr, Joe Taylor, Troy Luhman, Eric Luhman, Clarence Ng, Ricky Wang, and Aditya Ramesh. Video generation models as world simulators. *OpenAI technical reports*, 2024. 3

[12] Emanuele Bugliarello, H Hernan Moraldo, Ruben Villegas, Mohammad Babaeizadeh, Mohammad Taghi Saffar, Han Zhang, Dumitru Erhan, Vittorio Ferrari, Pieter-Jan Kindermans, and Paul Voigtlaender. StoryBench: a multifaceted benchmark for continuous story visualization. *NeurIPS*, 2024. 2, 6, 7

[13] Haoxin Chen, Menghan Xia, Yingqing He, Yong Zhang, Xiaodong Cun, Shaoshu Yang, Jinbo Xing, Yaofang Liu, Qifeng Chen, Xintao Wang, Chao Weng, and Ying Shan. Videocrafter1: Open diffusion models for high-quality video generation. *arXiv preprint arXiv:2310.19512*, 2023. 1, 2, 3

[14] Haoxin Chen, Yong Zhang, Xiaodong Cun, Menghan Xia, Xintao Wang, Chao Weng, and Ying Shan. Videocrafter2: Overcoming data limitations for high-quality video diffusion models. In *CVPR*, 2024. 1, 2

[15] Lin Chen, Xilin Wei, Jinsong Li, Xiaoyi Dong, Pan Zhang, Yuhang Zang, Zehui Chen, Haodong Duan, Bin Lin, Zhenyu Tang, et al. ShareGPT4Video: Improving video understanding and generation with better captions. *NeurIPS*, 2024. 3

[16] Shouyuan Chen, Sherman Wong, Liangjian Chen, and Yuandong Tian. Extending context window of large language models via positional interpolation. *arXiv preprint arXiv:2306.15595*, 2023. 5

[17] Shoufa Chen, Mengmeng Xu, Jiawei Ren, Yuren Cong, Sen He, Yanping Xie, Animesh Sinha, Ping Luo, Tao Xiang, and Juan-Manuel Perez-Rua. GenTron: Diffusion transformers for image and video generation. In *CVPR*, 2024. 3

[18] Tsai-Shien Chen, Aliaksandr Siarohin, Willi Menapace, Ekaterina Deyneka, Hsiang-wei Chao, Byung Eun Jeon, Yuwei Fang, Hsin-Ying Lee, Jian Ren, Ming-Hsuan Yang, et al. Panda-70M: Captioning 70m videos with multiple cross-modality teachers. In *CVPR*, 2024. 3, 5, 6, 2

[19] Tri Dao, Dan Fu, Stefano Ermon, Atri Rudra, and Christopher Ré. FlashAttention: Fast and memory-efficient exact attention with io-awareness. *NeurIPS*, 2022. 3

[20] Dave Epstein, Boyuan Chen, and Carl Vondrick. Oops! predicting unintentional action in video. In *CVPR*, 2020. 2

[21] 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. 4

[22] Wang Fu-Yun, Huang Zhaoyang, Ma Qiang, Song Guanglu, Lu Xudong, Bian Weikang, Li Yijin, Liu Yu, and Li Hongsheng. ZoLA: Zero-shot creative long animation generation with short video model. In *ECCV*, 2024. 2, 3

[23] Peng Gao, Le Zhuo, Ziyi Lin, Dongyang Liu, Ruoyi Du, Xu Luo, Longtian Qiu, Yuhang Zhang, et al. Lumina-t2x: Transforming text into any modality, resolution, and duration via flow-based large diffusion transformers. *arXiv preprint arXiv:2405.05945*, 2024. 3

[24] Yuan Gong, Youxin Pang, Xiaodong Cun, Menghan Xia, Yingqing He, Haoxin Chen, Longyue Wang, Yong Zhang, Xintao Wang, Ying Shan, et al. TaleCrafter: Interactive story visualization with multiple characters. *arXiv preprint arXiv:2305.18247*, 2023. 3

[25] Jiaxi Gu, Shicong Wang, Haoyu Zhao, Tianyi Lu, Xing Zhang, Zuxuan Wu, Songcen Xu, Wei Zhang, Yu-Gang Jiang, and Hang Xu. Reuse and diffuse: Iterative denoising for text-to-video generation. *arXiv preprint arXiv:2309.03549*, 2023. 3

[26] Yuwei Guo, Ceyuan Yang, Anyi Rao, Yaohui Wang, Yu Qiao, Dahua Lin, and Bo Dai. Animatediff: Animate your

personalized text-to-image diffusion models without specific tuning. In *ICLR*, 2024. 1, 2

[27] Agrim Gupta, Lijun Yu, Kihyuk Sohn, Xiuye Gu, Meera Hahn, Li Fei-Fei, Irfan Essa, Lu Jiang, and José Lezama. Photorealistic video generation with diffusion models. *arXiv preprint arXiv:2312.06662*, 2023. 3

[28] Xuan He, Dongfu Jiang, Ge Zhang, Max Ku, Achint Soni, Sherman Siu, Haonan Chen, Abhranil Chandra, Ziyan Jiang, Aaran Arulraj, et al. VideoScore: Building automatic metrics to simulate fine-grained human feedback for video generation. In *EMNLP*, 2024. 6, 3

[29] Yingqing He, Tianyu Yang, Yong Zhang, Ying Shan, and Qifeng Chen. Latent video diffusion models for high-fidelity video generation with arbitrary lengths. *arXiv preprint arXiv:2211.13221*, 2022. 2, 3

[30] Yingqing He, Menghan Xia, Haoxin Chen, Xiaodong Cun, Yuan Gong, Jinbo Xing, Yong Zhang, Xintao Wang, Chao Weng, Ying Shan, et al. Animate-A-Story: Storytelling with retrieval-augmented video generation. *arXiv preprint arXiv:2307.06940*, 2023. 3

[31] Roberto Henschel, Levon Khachatryan, Daniil Hayrapetyan, Hayk Poghosyan, Vahram Tadevosyan, Zhangyang Wang, Shant Navasardyan, and Humphrey Shi. StreamingT2V: Consistent, dynamic, and extendable long video generation from text. *arXiv preprint arXiv:2403.14773*, 2024. 2, 3, 8

[32] Jack Hessel, Ari Holtzman, Maxwell Forbes, Ronan Le Bras, and Yejin Choi. Clipscore: A reference-free evaluation metric for image captioning. In *EMNLP*, 2021. 6, 3

[33] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *NeurIPS*, 2017. 6, 3

[34] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022. 6, 4

[35] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *NeurIPS*, 2020. 2

[36] Jonathan Ho, William Chan, Chitwan Saharia, Jay Whang, Ruiqi Gao, Alexey Gritsenko, Diederik P Kingma, Ben Poole, Mohammad Norouzi, David J Fleet, et al. Imagen video: High definition video generation with diffusion models. *arXiv preprint arXiv:2210.02303*, 2022. 1, 2

[37] Jonathan Ho, Tim Salimans, Alexey Gritsenko, William Chan, Mohammad Norouzi, and David J Fleet. Video diffusion models. *NeurIPS*, 2022. 1, 2

[38] Gabriel Huang, Bo Pang, Zhenhai Zhu, Clara Rivera, and Radu Soricut. Multimodal pretraining for dense video captioning. *arXiv preprint arXiv:2011.11760*, 2020. 2

[39] 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*, 2024. 2, 6, 7

[40] Yuming Jiang, Tianxing Wu, Shuai Yang, Chenyang Si, Dahua Lin, Yu Qiao, Chen Change Loy, and Ziwei Liu. VideoBooth: Diffusion-based video generation with image prompts. In *CVPR*, 2024. 8

[41] Xuan Ju, Yiming Gao, Zhaoyang Zhang, Ziyang Yuan, Xintao Wang, Ailing Zeng, Yu Xiong, Qiang Xu, and Ying Shan. MiraData: A large-scale video dataset with long durations and structured captions. *arXiv preprint arXiv:2407.06358*, 2024. 3, 5

[42] Levon Khachatryan, Andranik Movsisyan, Vahram Tadevosyan, Roberto Henschel, Zhangyang Wang, Shant Navasardyan, and Humphrey Shi. Text2video-zero: Text-to-image diffusion models are zero-shot video generators. In *CVPR*, 2023. 2

[43] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013. 3

[44] Ranjay Krishna, Kenji Hata, Frederic Ren, Li Fei-Fei, and Juan Carlos Niebles. Dense-captioning events in videos. In *ICCV*, 2017. 2, 5

[45] Tuomas Kynkäänniemi, Miika Aittala, Tero Karras, Samuli Laine, Timo Aila, and Jaakko Lehtinen. Applying guidance in a limited interval improves sample and distribution quality in diffusion models. *arXiv preprint arXiv:2404.07724*, 2024. 4

[46] PKU-Yuan Lab and Tuzhan AI etc. Open-Sora-Plan, 2024. 3, 6, 1

[47] Yuheng Li, Haotian Liu, Qingyang Wu, Fangzhou Mu, Jianwei Yang, Jianfeng Gao, Chunyuan Li, and Yong Jae Lee. GLIGEN: Open-set grounded text-to-image generation. In *CVPR*, 2023. 2, 4, 8

[48] Yunxin Li, Haoyuan Shi, Baotian Hu, Longyue Wang, Jiashun Zhu, Jinyi Xu, Zhen Zhao, and Min Zhang. Anim-Director: A large multimodal model powered agent for controllable animation video generation. In *SIGGRAPH Asia Conference Track*, 2024. 3

[49] Long Lian, Baifeng Shi, Adam Yala, Trevor Darrell, and Boyi Li. Llm-grounded video diffusion models. In *ICLR*, 2024. 2, 5, 8

[50] Han Lin, Abhay Zala, Jaemin Cho, and Mohit Bansal. VideoDirectorGPT: Consistent multi-scene video generation via llm-guided planning. In *COLM*, 2024. 2, 3, 5, 8

[51] Yaron Lipman, Ricky TQ Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. In *ICLR*, 2023. 2, 3, 4

[52] Xingchao Liu, Chengyue Gong, et al. Flow straight and fast: Learning to generate and transfer data with rectified flow. In *ICLR*, 2023. 3, 4

[53] Xiaoran Liu, Hang Yan, Shuo Zhang, Chenxin An, Xipeng Qiu, and Dahua Lin. Scaling laws of rope-based extrapolation. In *ICLR*, 2024. 1

[54] Fuchen Long, Zhaofan Qiu, Ting Yao, and Tao Mei. VideoStudio: Generating consistent-content and multi-scene videos. In *ECCV*, 2024. 2, 3, 6, 8

[55] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *ICLR*, 2019. 6, 4

[56] Zeyu Lu, ZiDong Wang, Di Huang, Chengyue Wu, Xihui Liu, Wanli Ouyang, and LEI BAI. FiT: Flexible vision transformer for diffusion model. In *ICML*, 2024. 3

[57] Xin Ma, Yaohui Wang, Gengyun Jia, Xinyuan Chen, Ziwei Liu, Yuan-Fang Li, Cunjian Chen, and Yu Qiao. Latte:

Latent diffusion transformer for video generation. *arXiv preprint arXiv:2401.03048*, 2024. 3

[58] Adyasha Maharana and Mohit Bansal. Integrating visuospatial, linguistic and commonsense structure into story visualization. In *EMNLP*, 2021. 3

[59] Adyasha Maharana, Darryl Hannan, and Mohit Bansal. Improving generation and evaluation of visual stories via semantic consistency. In *NAACL*, 2021.

[60] Adyasha Maharana, Darryl Hannan, and Mohit Bansal. StoryDALL-E: Adapting pretrained text-to-image transformers for story continuation. In *ECCV*, 2022. 3

[61] Willi Menapace, Aliaksandr Siarohin, Ivan Skorokhodov, Ekaterina Deyneka, Tsai-Shien Chen, Anil Kag, Yuwei Fang, Aleksei Stoliar, Elisa Ricci, Jian Ren, et al. Snap video: Scaled spatiotemporal transformers for text-to-video synthesis. *CVPR*, 2024. 3

[62] Bolin Ni, Houwen Peng, Minghao Chen, Songyang Zhang, Gaofeng Meng, Jianlong Fu, Shiming Xiang, and Haibin Ling. Expanding language-image pretrained models for general video recognition. In *ECCV*, 2022. 3

[63] Gyeongrok Oh, Jaehwan Jeong, Sieun Kim, Wonmin Byeon, Jinkyu Kim, Sungwoong Kim, and Sangpil Kim. MEVG: Multi-event video generation with text-to-video models. In *ECCV*, 2024. 1, 3, 6

[64] Xichen Pan, Pengda Qin, Yuhong Li, Hui Xue, and Wenhu Chen. Synthesizing coherent story with auto-regressive latent diffusion models. In *WACV*, 2024. 3

[65] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. PyTorch: An imperative style, high-performance deep learning library. *NeurIPS*, 2019. 4

[66] William Peebles and Saining Xie. Scalable diffusion models with transformers. *ICCV*, 2023. 2, 3

[67] Bowen Peng, Jeffrey Quesnelle, Honglu Fan, and Enrico Shippole. YaRN: Efficient context window extension of large language models. In *ICLR*, 2024. 1

[68] 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. *arXiv preprint arXiv:2307.01952*, 2023. 5, 2

[69] Adam Polyak, Amit Zohar, Andrew Brown, Andros Tjandra, Animesh Sinha, Ann Lee, Apoorv Vyas, Bowen Shi, Chih-Yao Ma, Ching-Yao Chuang, et al. Movie gen: A cast of media foundation models. *arXiv preprint arXiv:2410.13720*, 2024. 1, 3

[70] Haonan Qiu, Menghan Xia, Yong Zhang, Yingqing He, Xintao Wang, Ying Shan, and Ziwei Liu. FreeNoise: Tuning-free longer video diffusion via noise rescheduling. In *ICLR*, 2024. 3, 8

[71] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *ICML*, 2021. 3

[72] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *JMLR*, 2020. 3

[73] Tanzila Rahman, Hsin-Ying Lee, Jian Ren, Sergey Tulyakov, Shweta Mahajan, and Leonid Sigal. Make-A-Story: Visual memory conditioned consistent story generation. In *CVPR*, 2023. 3

[74] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *CVPR*, 2022. 3

[75] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-Net: Convolutional networks for biomedical image segmentation. In *MICCAI*, 2015. 3

[76] Abhishek Sharma, Adams Yu, Ali Razavi, Andeep Toor, Andrew Pierson, Ankush Gupta, Austin Waters, Daniel Tanis, Dumitru Erhan, Eric Lau, Eleni Shaw, Gabe Barth-Maron, Greg Shaw, Han Zhang, Henna Nandwani, Hernan Moraldo, Hyunjik Kim, Irina Blok, Jakob Bauer, Jeff Donahue, Junyoung Chung, Kory Mathewson, Kurtis David, Lasse Espeholt, Marc van Zee, Matt McGill, Medhini Narasimhan, Miaosen Wang, Mikołaj Bińkowski, Mohammad Babaeizadeh, Mohammad Taghi Saffar, Nick Pezzotti, Pieter-Jan Kindermans, Poorva Rane, Rachel Hornung, Robert Riachi, Ruben Villegas, Rui Qian, Sander Dieleman, Serena Zhang, Serkan Cabi, Shixin Luo, Shlomi Fruchter, Signe Nørly, Srivatsan Srinivasan, Tobias Pfaff, Tom Hume, Vikas Verma, Weizhe Hua, William Zhu, Xinchen Yan, Xinyu Wang, Yelin Kim, Yuqing Du, and Yutian Chen. Veo, 2024. 3

[77] Uriel Singer, Adam Polyak, Thomas Hayes, Xi Yin, Jie An, Songyang Zhang, Qiyuan Hu, Harry Yang, Oron Ashual, Oran Gafni, et al. Make-a-video: Text-to-video generation without text-video data. In *ICLR*, 2023. 2

[78] Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *ICML*, 2015. 2

[79] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. In *ICLR*, 2021. 6, 3

[80] Tomáš Souček and Jakub Lokoč. TransNet V2: An effective deep network architecture for fast shot transition detection. *arXiv preprint arXiv:2008.04838*, 2020. 6, 2, 3

[81] Jianlin Su, Murtadha Ahmed, Yu Lu, Shengfeng Pan, Wen Bo, and Yunfeng Liu. Roformer: Enhanced transformer with rotary position embedding. *Neurocomputing*, 2024. 2, 3, 1

[82] Genmo Team. Mochi, 2024. 2, 6, 9, 10

[83] Sergey Tulyakov, Ming-Yu Liu, Xiaodong Yang, and Jan Kautz. MoCoGAN: Decomposing motion and content for video generation. In *CVPR*, 2018. 2, 4

[84] Thomas Unterthiner, Sjoerd van Steenkiste, Karol Kurach, Raphael Marinier, Marcin Michalski, and Sylvain Gelly. Towards accurate generative models of video: A new metric & challenges. *arXiv preprint arXiv:1812.01717*, 2018. 6, 3

[85] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser,

and Illia Polosukhin. Attention is all you need. *NeurIPS*, 2017. 3

[86] Ruben Villegas, Jimei Yang, Seunghoon Hong, Xunyu Lin, and Honglak Lee. Decomposing motion and content for natural video sequence prediction. In *ICLR*, 2017. 2, 4

[87] Ruben Villegas, Mohammad Babaeizadeh, Pieter-Jan Kindermans, Hernan Moraldo, Han Zhang, Mohammad Taghi Saffar, Santiago Castro, Julius Kunze, and Dumitru Erhan. Phenaki: Variable length video generation from open domain textual descriptions. In *ICLR*, 2022. 1, 3, 7

[88] Fu-Yun Wang, Wenshuo Chen, Guanglu Song, Han-Jia Ye, Yu Liu, and Hongsheng Li. Gen-L-Video: Multi-text to long video generation via temporal co-denoising. *arXiv preprint arXiv:2305.18264*, 2023. 1, 3, 8

[89] Jiuniu Wang, Hangjie Yuan, Dayou Chen, Yingya Zhang, Xiang Wang, and Shiwei Zhang. Modelscope text-to-video technical report. *arXiv preprint arXiv:2308.06571*, 2023. 2

[90] Jiawei Wang, Yuchen Zhang, Jiaxin Zou, Yan Zeng, Guoqiang Wei, Liping Yuan, and Hang Li. Boximator: Generating rich and controllable motions for video synthesis. *arXiv preprint arXiv:2402.01566*, 2024. 4, 8

[91] Weiyao Wang, Matt Feiszli, Heng Wang, and Du Tran. Unidentified video objects: A benchmark for dense, open-world segmentation. In *ICCV*, 2021. 2

[92] Xudong Wang, Trevor Darrell, Sai Saketh Rambhatla, Rohit Girdhar, and Ishan Misra. InstanceDiffusion: Instance-level control for image generation. In *CVPR*, 2024. 4

[93] Xiang Wang, Hangjie Yuan, Shiwei Zhang, Dayou Chen, Jiuniu Wang, Yingya Zhang, Yujun Shen, Deli Zhao, and Jingren Zhou. VideoComposer: Compositional video synthesis with motion controllability. *NeurIPS*, 2024. 2

[94] Yi Wang, Yinan He, Yizhuo Li, Kunchang Li, Jiashuo Yu, Xin Ma, Xinhao Li, Guo Chen, Xinyuan Chen, Yaohui Wang, et al. InternVid: A large-scale video-text dataset for multimodal understanding and generation. In *ICLR*, 2024. 5

[95] Jay Zhangjie Wu, Yixiao Ge, Xintao Wang, Stan Weixian Lei, Yuchao Gu, Yufei Shi, Wynne Hsu, Ying Shan, Xiaohu Qie, and Mike Zheng Shou. Tune-a-video: One-shot tuning of image diffusion models for text-to-video generation. In *ICCV*, 2023. 3

[96] Jinbo Xing, Menghan Xia, Yong Zhang, Haoxin Chen, Xintao Wang, Tien-Tsin Wong, and Ying Shan. DynamiCrafter: Animating open-domain images with video diffusion priors. In *ECCV*, 2024. 1, 7

[97] Tianwei Xiong, Yuqing Wang, Daquan Zhou, Zhijie Lin, Jiashi Feng, and Xihui Liu. Lvd-2m: A long-take video dataset with temporally dense captions. *NeurIPS*, 2024. 3, 7

[98] Hongwei Xue, Tiankai Hang, Yanhong Zeng, Yuchong Sun, Bei Liu, Huan Yang, Jianlong Fu, and Baining Guo. Advancing high-resolution video-language representation with large-scale video transcriptions. In *CVPR*, 2022. 6, 2

[99] Antoine Yang, Arsha Nagrani, Ivan Laptev, Josef Sivic, and Cordelia Schmid. VidChapters-7M: Video chapters at scale. *NeurIPS*, 2023. 2

[100] Zhuoyi Yang, Jiayan Teng, Wendi Zheng, Ming Ding, Shiyu Huang, Jiazheng Xu, Yuanming Yang, Wenyi Hong, Xiaohan Zhang, Guanyu Feng, et al. CogVideoX: Text-to-video diffusion models with an expert transformer. *arXiv preprint arXiv:2408.06072*, 2024. 2, 3, 6, 9, 10

[101] Lijun Yu, José Lezama, Nitesh B Gundavarapu, Luca Versari, Kihyuk Sohn, David Minnen, Yong Cheng, Agrim Gupta, Xiuye Gu, Alexander G Hauptmann, et al. Language model beats diffusion–tokenizer is key to visual generation. *arXiv preprint arXiv:2310.05737*, 2023. 3

[102] Canyu Zhao, Mingyu Liu, Wen Wang, Jianlong Yuan, Hao Chen, Bo Zhang, and Chunhua Shen. MovieDreamer: Hierarchical generation for coherent long visual sequence. *arXiv preprint arXiv:2407.16655*, 2024. 3

[103] Zangwei Zheng, Xiangyu Peng, Tianji Yang, Chenhui Shen, Shenggui Li, Hongxin Liu, Yukun Zhou, Tianyi Li, and Yang You. Open-Sora: Democratizing efficient video production for all, 2024. 3, 6, 2

[104] Daquan Zhou, Weimin Wang, Hanshu Yan, Weiwei Lv, Yizhe Zhu, and Jiashi Feng. MagicVideo: Efficient video generation with latent diffusion models. *arXiv preprint arXiv:2211.11018*, 2022. 3

[105] Luowei Zhou, Chenliang Xu, and Jason Corso. Towards automatic learning of procedures from web instructional videos. In *AAAI*, 2018. 5, 2

[106] Yupeng Zhou, Daquan Zhou, Ming-Ming Cheng, Jiashi Feng, and Qibin Hou. StoryDiffusion: Consistent self-attention for long-range image and video generation. *arXiv preprint arXiv:2405.01434*, 2024. 3

[107] Zhongyang Zhu and Jie Tang. CogCartoon: Towards practical story visualization. *IJCV*, 2024. 3

Mind the Time: Temporally-Controlled Multi-Event Video Generation

Supplementary Material

We highly encourage the readers to check out our project page for video results of baselines and MinT.

A. Details on Rotary Position Embedding

A.1. Derivation of RoPE

We detail the derivation conducted in Sec. 3.1 of the main paper. Our derivation mostly follows [53, 67, 81] and only provides an intuitive motivation for our method. We refer readers to their papers for more rigorous results.

Given a query vector $\boldsymbol{q}_n = \left[q_n^{(0)}, \cdots, q_n^{(d-1)}\right] \in \mathbb{R}^d$ at index n and a key vector $\boldsymbol{k}_m = \left[k_m^{(0)}, \cdots, k_m^{(d-1)}\right] \in \mathbb{R}^d$ at index m, to apply RoPE, it first groups every two elements in them, and make them complex numbers as:

$$\begin{aligned} \bar{\boldsymbol{q}}_n &= \left[\bar{q}_n^{(0)}, \cdots, \bar{q}_n^{(d/2-1)}\right], \quad \bar{q}_n^{(l)} = q_n^{(2l)} + i q_n^{(2l+1)}, \\ \bar{\boldsymbol{k}}_m &= \left[\bar{k}_m^{(0)}, \cdots, \bar{k}_m^{(d/2-1)}\right], \quad \bar{k}_m^{(l)} = k_m^{(2l)} + i k_m^{(2l+1)}. \end{aligned} \tag{1}$$

Then, RoPE rotates each complex number by an angle θ_l, which is achieved as element-wise multiplication:

$$\tilde{\boldsymbol{q}}_n = \bar{\boldsymbol{q}}_n \odot e^{in\boldsymbol{\theta}}, \quad \tilde{\boldsymbol{k}}_m = \bar{\boldsymbol{k}}_m \odot e^{im\boldsymbol{\theta}}, \tag{2}$$

where $\boldsymbol{\theta}$ is determined by the position l of each element in a vector. We follow prior works [46, 81] to use:

$$\boldsymbol{\theta} = \left[\theta_0, \cdots, \theta_{d/2-1}\right], \quad \theta_l = 10000^{-2l/d}. \tag{3}$$

Eq. (3) indicates that each θ_l is a fixed value, and thus the rotation results in Eq. (2) is only decided by the vectors' index n and m. This is why in the main paper, we only consider a single θ_{base} instead of $\boldsymbol{\theta}$ for different elements.

We can now calculate the attention between $\tilde{\boldsymbol{q}}_n$ and $\tilde{\boldsymbol{k}}_m$:

$$\begin{aligned} A_{n,m} &= \operatorname{Re}\left[\langle \tilde{\boldsymbol{q}}_n, \tilde{\boldsymbol{k}}_m \rangle\right] \\ &= \operatorname{Re}\left[(\bar{\boldsymbol{q}}_n e^{in\boldsymbol{\theta}}) \cdot (\bar{\boldsymbol{k}}_m e^{im\boldsymbol{\theta}})^*\right] \\ &= \operatorname{Re}\left[\sum_{l=0}^{d/2-1} (\bar{q}_n^{(l)} e^{in\theta_l})(\bar{k}_m^{(l)*} e^{-im\theta_l})\right] \\ &= \operatorname{Re}\left[\sum_{l=0}^{d/2-1} \bar{q}_n^{(l)} \bar{k}_m^{(l)*} e^{i(n-m)\theta_l}\right] \\ &= \sum_{l=0}^{d/2-1} \left(q_n^{(2l)} k_m^{(2l)} + q_n^{(2l+1)} k_m^{(2l+1)}\right) \cos\left((n-m)\theta_l\right) + \\ &\qquad \left(q_n^{(2l)} k_m^{(2l+1)} - q_n^{(2l+1)} k_m^{(2l)}\right) \sin\left((n-m)\theta_l\right). \end{aligned} \tag{4}$$

Since we are interested in the bias introduced by RoPE in attention, we assume all queries $\boldsymbol{q}_n$ and all keys $\boldsymbol{k}_m$ are the same, so that their attention values without RoPE is the same. Empirically, we find that query and key vectors indeed have similar values in our DiT due to the use of Layer Normalization [5]. Thanks to the periodic property of $\sin(\cdot)$ and $\cos(\cdot)$, from Eq. (4), we have $A_{n,m} = A_{m,n}$, i.e., , the attention bias between $\boldsymbol{q}_n$ and $\boldsymbol{k}_m$ is only affected by the absolute distance between the two vectors, $|n-m|$.

The original RoPE paper [81] proves that the upper bound of $A_{n,m}$ decays monotonically with the distance $|n-m|$ until around 40. Since the RoPE used in the temporal cross-attention layer only encodes vectors using the temporal frame index, and our video DiT is trained on video tokens with up to around 50 frames, we roughly preserve the monotonicity of RoPE. As we will see in Appendix A.3, while there are some fluctuations of $A_{n,m}$ in the long range, the long-term decay makes their values significantly low.

A.2. Proof of the Property of ReRoPE

In Sec. 3.2 of the main paper, we propose to rescale all events to a fixed length L. For a timestamp t lying in the n-th event, we transform it as:

$$\begin{aligned} \tilde{t} &= \frac{(t - t_n^{\text{start}})L}{t_n^{\text{end}} - t_n^{\text{start}}} + (n-1)L, \quad \text{s.t. } t_n^{\text{start}} \le t \le t_n^{\text{end}}, \\ \tilde{t}_n^{\text{mid}} &= L/2 + (n-1)L. \end{aligned} \tag{5}$$

After transformation, the distance between a video token in the n-th event and the middle timestamps of this event is:

$$\left|\tilde{t} - \tilde{t}_n^{\text{mid}}\right| = \left|\frac{t - t_n^{\text{start}}}{t_n^{\text{end}} - t_n^{\text{start}}} - \frac{1}{2}\right| L. \tag{6}$$

Next, we prove that it satisfies the three desired properties of the temporal cross-attention:

(i) *For video tokens within the time span of an event, they should attend the most to the text embedding of this event.*

Proof For $t_n^{\text{start}} \le t \le t_n^{\text{end}}$, we have:

$$-\frac{1}{2} \le \left(\frac{t - t_n^{\text{start}}}{t_n^{\text{end}} - t_n^{\text{start}}} - \frac{1}{2}\right) \le \frac{1}{2}, \tag{7}$$

thus, $\left|\tilde{t} - \tilde{t}_n^{\text{mid}}\right| \le L/2$. For any m-th event with $m \ne n$, its distance to this video token is:

$$\left|\tilde{t} - \tilde{t}_m^{\text{mid}}\right| = \left|\left(\frac{t - t_n^{\text{start}}}{t_n^{\text{end}} - t_n^{\text{start}}} - \frac{1}{2}\right) + (n-m)\right| L. \tag{8}$$

Since $|n-m| \ge 1$, we get:

$$\left|\left(\frac{t - t_n^{\text{start}}}{t_n^{\text{end}} - t_n^{\text{start}}} - \frac{1}{2}\right) + (n-m)\right| \ge \frac{1}{2}. \tag{9}$$

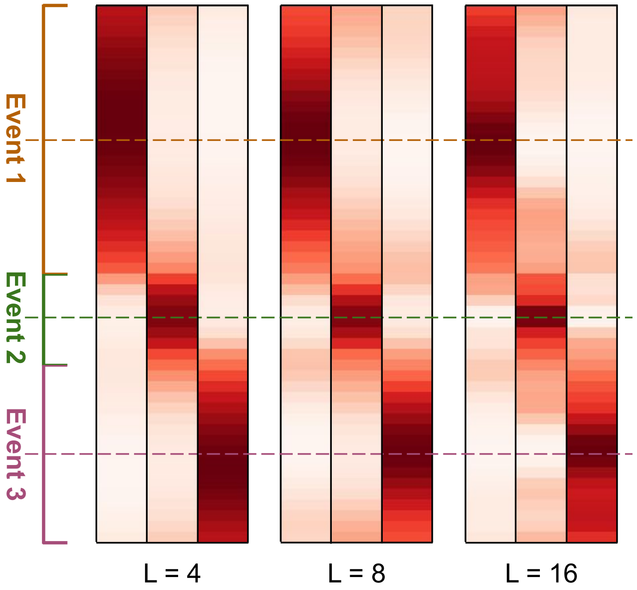


Figure 9. **Comparison of ReRoPE with different rescaling length L.** We use the same random vector for video tokens and text embeddings to only visualize the bias introduced by positional encoding. We visualize the case where videos have a temporal dimension of 50, and there are three temporal captions.

Therefore, we have:

$$\left|\tilde{t} - \tilde{t}_m^{\text{mid}}\right| \geq L/2 \geq \left|\tilde{t} - \tilde{t}_n^{\text{mid}}\right|, \ \forall\, m \neq n. \tag{10}$$

Since RoPE attention decays monotonically with the distance, we reach the property.

(ii) *For an event, the attention weight should peak with the video token at the midpoint of its time span, and then decrease towards the boundary of the event.*

Proof When a video token is at the midpoint of an event, we have $\tilde{t} - \tilde{t}_n^{\text{mid}} = 0$. Thus, the attention weight will be the highest. In addition, Eq. (6) increases when t goes from t_n^{mid} to t_n^{start} or t_n^{end}, leading to a decreased weight.

(iii) *The video token at the transition point between two events should attend equally to their text embeddings.*

Proof For $t = t_n^{\text{start}}$ or t_n^{end}, we always have the distance $\left|\tilde{t} - \tilde{t}_n^{\text{mid}}\right| = L/2$. Thus, the attention value is the same for video tokens at event borders. This is only possible in ReRoPE as we rescale all events to have the same length.

A.3. Visualizations of ReRoPE

In Sec. 4.5 of the main paper, we show that using different rescaling length L in ReRoPE leads to similar results. Fig. 9 visualizes the cross-attention map using $L = 4$, 8, and 16. The three attention maps are indeed similar, which explains why the performances are close. We also notice that with a higher L, the attention map of each event becomes more concentrated. It would be an interesting direction to study its effect in depth, which we leave for future work.

B. Detailed Experimental Setup

In this section, we provide full details on the datasets, baselines, evaluation settings, and the training and inference implementation details of our model.

B.1. Training Data

Before this work, there are mainly two types of video datasets that annotate open-set event captions and their precise timestamps. One such field is dense video captioning [38, 44, 105]. However, these datasets are limited in scale (usually fewer than 10k videos), which makes it impossible to fine-tune a large-scale video generator. Another field is video chaptering [99]. However, the temporal captions here are high-level chapter segmentation, where each annotated event is usually longer than one minute. This is too long for current video diffusion models to be trained on.

Since our model requires large-scale and fine-grained video event annotations, we manually source videos from existing datasets [18, 98] and annotate them, resulting in around 200k videos. To condition the model on scene cuts, we run TransNetV2 [80] to detect scene boundaries on annotated videos with a confidence threshold of 0.5.

Fig. 10 present some basic statistics of our dataset. While our training videos have varying lengths, the number of events per video and the average event length are similar, which makes model training easier.

Data processing. The training dataset contains videos of different lengths, resolutions, and aspect ratios. Following common practice [68, 103], we use data bucketing, which groups videos into a fixed set of sizes. Overall, we sample videos up to 512 resolution, and 10s during training. We pad to or subsample 4 temporal captions for batch training.

B.2. Evaluation Datasets

HoldOut. We randomly sample 2k videos from our training data as a holdout testing set. The prompts here are in-distribution with a minimum gap to training data.

StoryBench [12] consists of videos collected from DiDeMo [4], Oops [20], and UVO [91] datasets. It annotates each video with a background caption and one or more temporal captions similar to our format. We treat their background caption as the global caption in our setting, showing our model's generalization to out-of-distribution prompts. We filter out videos with only a single event, leading to around 3k testing samples.

VBench [39] is a comprehensive benchmark that tests different aspects of a video generation model. It has 16 evaluation dimensions, each with a carefully collected list of text prompts. Since we are interested in the dynamics of generated videos, we choose the *Dynamic Degree* dimension, which provides 72 prompts. Following the official evaluation protocol, we run each model to generate 5 videos using each prompt with 5 random seeds.

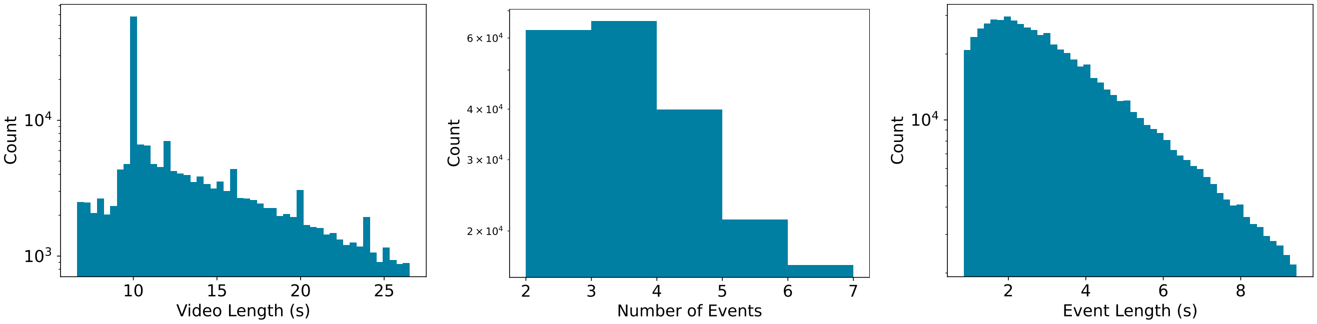


Figure 10. **Basic statistics of our training dataset.** We show the distribution of video length, the number of events per video, and the length of individual events. Most videos contain 2 to 4 events, and most events are under 5s.

B.3. Baselines

We only compare to methods that can generate smoothly connected events and have released their code.

MEVG [63] is the state-of-the-art multi-event video generation method. Given a sequence of event prompts, it generates the first video clip using the first event prompt. Then, to generate the next event, it runs DDIM inversion [79] to obtain the inverted noise latent of the previous clip, which is used to initialize the current noise latent. Then, when denoising the current latent, it also introduces several losses to enforce latent at adjacent frames to be similar. Original MEVG builds upon LVDM [29] and VideoCrafter [13] which are outdated. For a fair comparison, we re-implement it based on our base model. As far as we know, there is no prior work on inverting a rectified flow model, so we follow DDIM inversion to implement RF inversion which achieves similar results. To handle both global and temporal captions, we generate the first clip by concatenating the global caption and the first temporal caption. We keep other losses and hyper-parameters the same as in MEVG[2].

AutoReg. We fine-tune our base model to support initial frame conditioned video generation. The method is similar to MEVG, where we generate one event based on its own caption and the last frame of the previous clip.

Concat is a naive baseline that simply concatenates the global caption and all temporal captions to form a long prompt, and generates a video from it.

Remark. Since both MEVG and AutoReg are autoregressive methods, they can only generate fixed-length videos for each event. To enable comparison, we simply assume that the testing events all have the same duration when computing metrics. For Concat, it cannot separate the generation of different events. We thus assume all events are uniformly distributed in the generated video.

B.4. Evaluation Metrics

We identify three key aspects in multi-event text-to-video generation: visual quality, event text alignment, and event transition smoothness. We report common metrics such as FID [33], FVD [84] for visual quality, and per-frame CLIP-score [32, 71] for text alignment. We have tried more advanced metrics such as X-CLIP-score [62], but found it to perform similarly as CLIP-score.

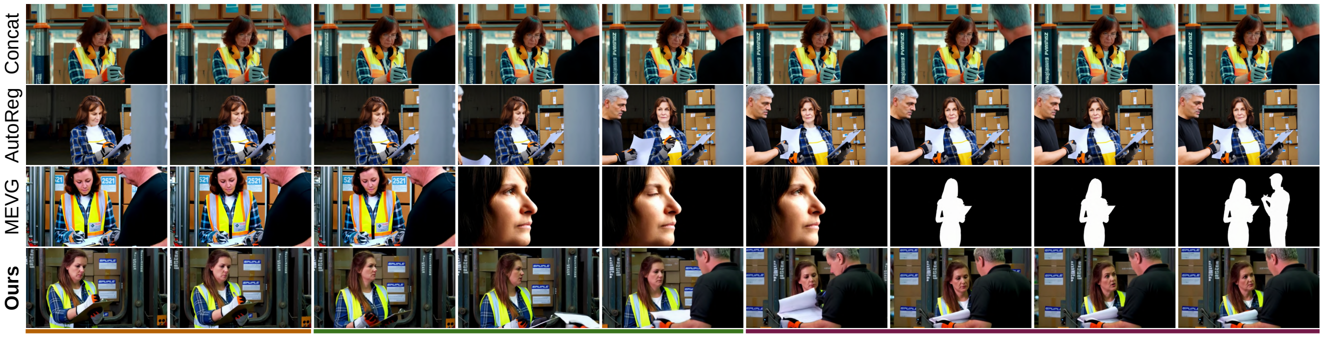
It is well-known that traditional automatic metrics are not aligned with human perceptions. Recent works show that fine-tuning multi-modal LLMs on human feedback data can lead to more human-aligned video quality assessment metrics [28]. We take the state-of-the-art method VideoScore which outputs five scores for a video. We use the *visual quality* and *dynamic degree* output for visual quality, the *text-to-video alignment* output for text alignment, and the *temporal consistency* output for event transition smoothness. We further run TransNetV2 [80] to compute the average number of cuts in generated videos to measure event transition smoothness.

For visual quality and event transition smoothness, we compute relevant metrics on the entire video. We have also computed the visual quality of each event, and found it to be positively correlated with video-level results. For text alignment, since we care about event generation, we take the start and end timestamps of each event, crop out a sub-clip from the generated video, and compute metrics between this sub-clip and the corresponding event prompt.

B.5. Implementation Details

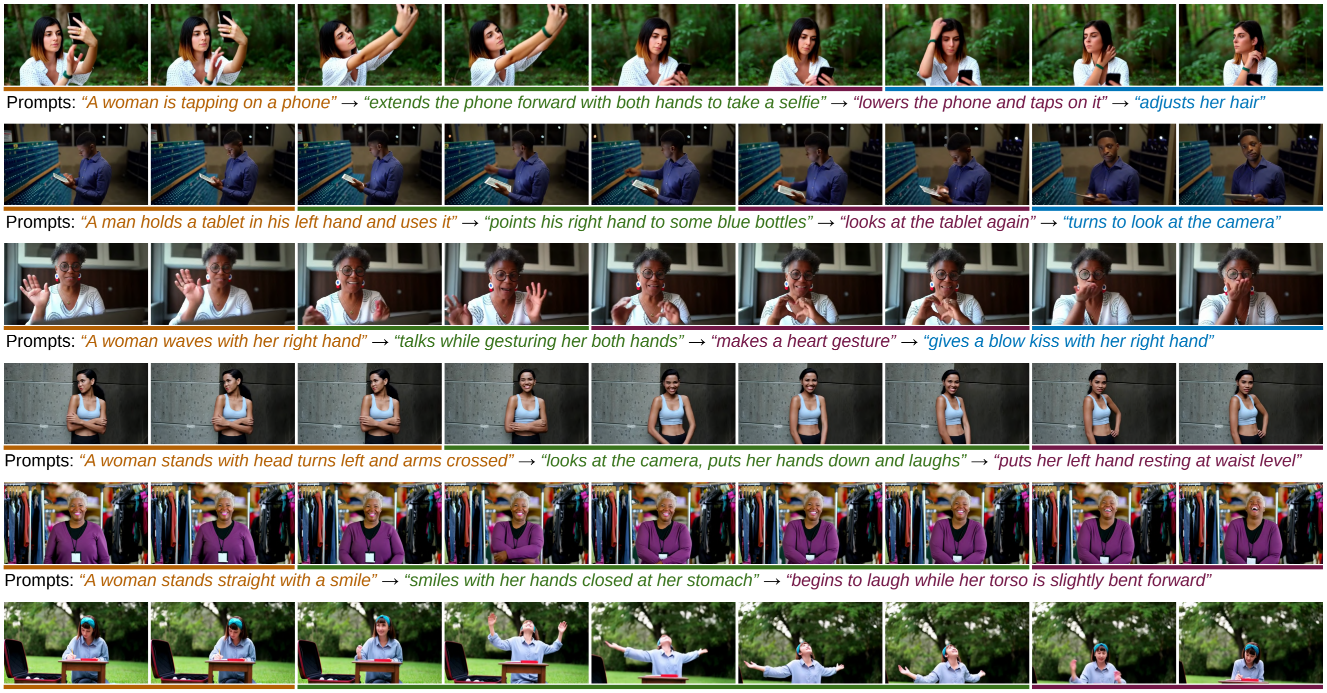
Base model. Our base text-to-video generator adopts the latent Diffusion Transformer framework [66]. It leverages a MAGVIT-v2 [101] as the autoencoder and a deep cascade of DiT blocks as the denoising backbone. The autoencoder is similar to the one in CogVideoX [100], which downsamples the spatial dimensions by 8× and the temporal dimension by 4×. Our backbone has 32 DiT blocks. Each block is similar to the one in Open-Sora [46], which consists of a 3D self-attention layer running on all video tokens, a cross-attention layer between video tokens and T5 text embeddings [72] of the input prompt, and an MLP. We do not use absolute positional encoding on video tokens. Instead, we apply RoPE in self-attention, which is factorized into spatial and temporal axes, similar to [46]. Finally, we use FlashAttention [19] in both self-attention and cross-attention.

[2] MEVG did not release the code at the time of paper submission. We obtain the official code from authors through private email communication.



Prompts: *"A woman is writing on a paper"* → *"She looks at the right as a man holding a clipboard is coming to her"* → *"They look at each other and discuss with the paper"*

Figure 11. **Qualitative comparisons of T2V.**



Prompts: *"A woman is tapping on a phone"* → *"extends the phone forward with both hands to take a selfie"* → *"lowers the phone and taps on it"* → *"adjusts her hair"*

Prompts: *"A man holds a tablet in his left hand and uses it"* → *"points his right hand to some blue bottles"* → *"looks at the tablet again"* → *"turns to look at the camera"*

Prompts: *"A woman waves with her right hand"* → *"talks while gesturing her both hands"* → *"makes a heart gesture"* → *"gives a blow kiss with her right hand"*

Prompts: *"A woman stands with head turns left and arms crossed"* → *"looks at the camera, puts her hands down and laughs"* → *"puts her left hand resting at waist level"*

Prompts: *"A woman stands straight with a smile"* → *"smiles with her hands closed at her stomach"* → *"begins to laugh while her torso is slightly bent forward"*

Prompts: *"A woman is writing something on a table"* → *"looks upwards with a smile and spreads her arms"* → *"resumes to write something on the table"*

Figure 12. **More T2V results from MinT.** Please see our project page for more results.

The base model adopts the rectified flow training objective [51, 52]. We follow Stable Diffusion 3 [21] to choose the sampling parameters for the diffusion process.

MinT model. We fine-tune MinT from the base model to enable temporal caption control. We copy weights from the original cross-attention layer to initialize our added temporal cross-attention layer to accelerate convergence, since both layers take in the same text modality. Following prior works [47], we introduce a scaling factor that is initialized as 0, and we pass it through a $\mathrm{Tanh}(\cdot)$ activation to multiply with the temporal cross-attention layer output. Such a design has been shown to stabilize model training.

Training. We use AdamW [55] to fine-tune the entire model with a batch size of 512 for 15k steps. We use a low learning rate of 1×10^{-5} for the pre-trained weights, and a higher one of 1×10^{-4} for the added weights. Both learning rates are linearly warmed up in the first 1k steps and stay constant. A gradient clipping of 0.05 is applied to stabilize training. To apply classifier-free guidance (CFG) [34], we randomly drop the text embedding of global and temporal captions (i.e., setting them as zeros) with a probability of 10%. Notice that when dropping the temporal captions, we drop all events together and also set the event timestamps to zeros. We implement our model using PyTorch [65] and conduct training on NVIDIA A100 GPUs.

Inference. We use the rectified flow sampler [52] with 256 sampling steps and a classifier-free guidance [34] scale of 8 to generate videos. We also use interval guidance [45] in CFG to mitigate the oversaturation issue, which only applies CFG between [25, 100] sampling steps. We have tried using separate CFG for global and temporal captions similar to in [10], but did not find it to improve results.

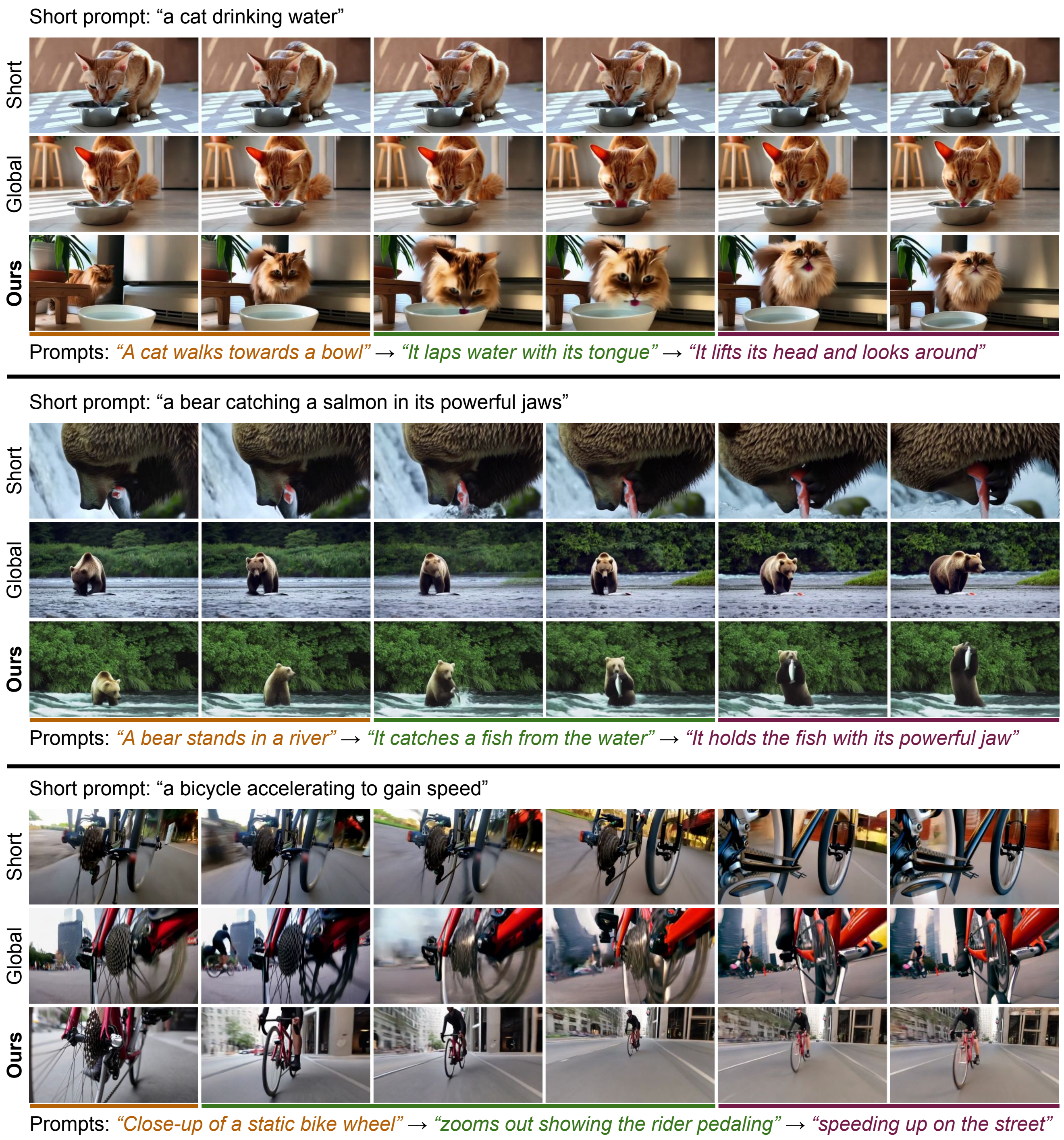

Short prompt: "a cat drinking water"
Short
Global
Ours
Prompts: "A cat walks towards a bowl" → "It laps water with its tongue" → "It lifts its head and looks around"
Short prompt: "a bear catching a salmon in its powerful jaws"
Short
Global
Ours
Prompts: "A bear stands in a river" → "It catches a fish from the water" → "It holds the fish with its powerful jaw"
Short prompt: "a bicycle accelerating to gain speed"
Short
Global
Ours
Prompts: "Close-up of a static bike wheel" → "zooms out showing the rider pedaling" → "speeding up on the street"

Figure 13. **Prompt enhancement results on VBench.** We can generate more interesting videos from a simple prompt. This highlights the flexible dynamics control ability brought by the temporal captions. Please see our project page for video results.

C. More Results

C.1. More Qualitative Results on T2V

Fig. 11 presents more qualitative comparisons with baselines. Concat only generates the woman writing on a paper while ignoring the subsequent events. AutoReg is able to synthesize a smooth transition between the first and the second event, but it fails to generate the third event. This is because conditioning on generated frames leads to video stagnation and results in frozen frames. MEVG generates each event well, but they are connected with abrupt shot transitions and completely different subjects. This is due to the free-form event captions we use, which change the subjects frequently. As a result, the inversion technique in

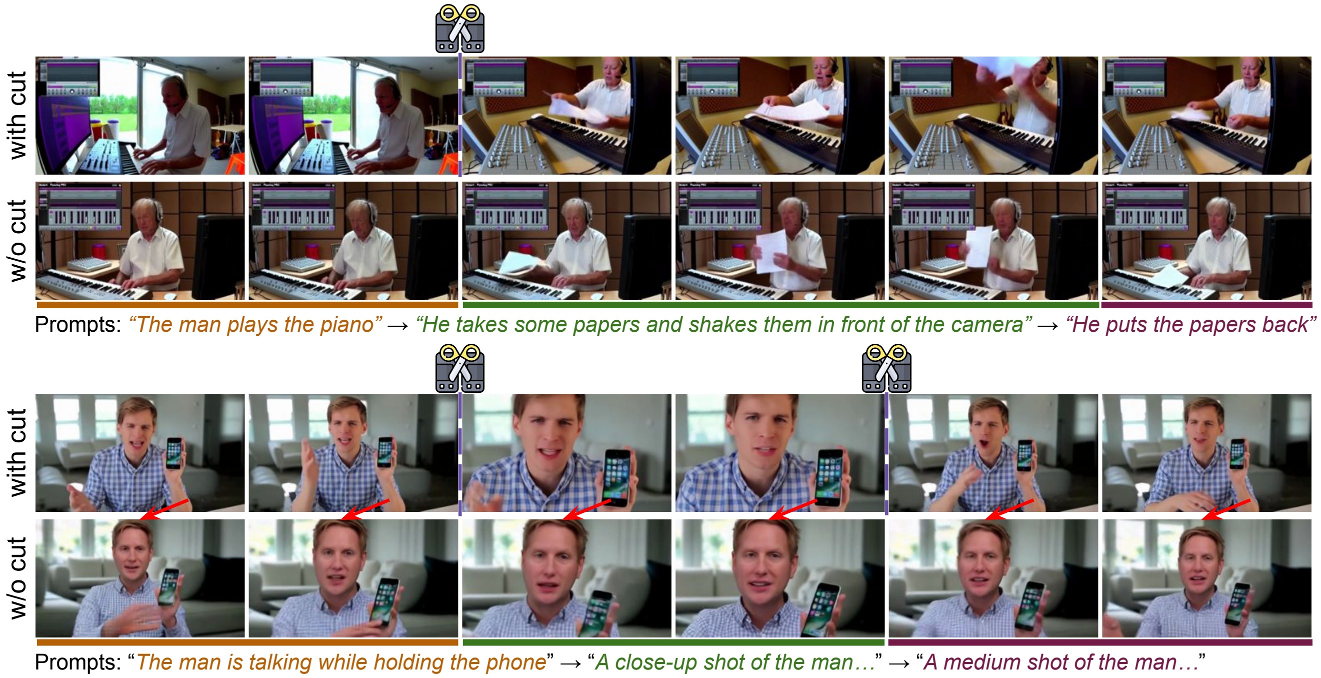

with cut
w/o cut
Prompts: "The man plays the piano" → "He takes some papers and shakes them in front of the camera" → "He puts the papers back"
with cut
w/o cut
Prompts: "The man is talking while holding the phone" → "A close-up shot of the man…" → "A medium shot of the man…"

Figure 14. **Generated videos with and without scene cut input.** In each example, the first row is generated by inputting the scene cut at the illustrated timestamps, while the second row is by zeroing out the scene cut input. When using the scene cut, the model is able to generate a shot transition at desired timestamps, while keeping the subject consistent. In the second example, the model generates smooth zoom-in and zoom-out effects when zeroing out scene cuts. Please see our project page for more results.

MEVG cannot preserve the subjects well. So far, there is no inversion method designed for rectified flow models. Overall, MinT is the only method that successfully generates all events with smooth transitions and consistent entities.

We show more qualitative results of MinT in Fig. 12. Human-related subjects are known to be challenging in visual generation tasks. Yet, the results demonstrate our flexible control of human action sequences and time lengths.

C.2. Prompt Enhancement

Our prompt enhancer is built upon GPT-4 [3] and can extend a short prompt to a detailed global caption and multiple temporal captions with reasonable event timestamps. We provide the instruction we used on our project page. It is inspired by recent works [54, 63] and uses in-context examples from our dataset for better performance.

We show more prompt enhancement results using VBench prompts in Fig. 13. Thanks to the powerful LLM, our prompt enhancer can extend a short prompt to reasonable sequential events, covering rich object motion and camera movement. MinT can then generate more interesting and "eventful" videos from the extended prompt. This highlights the unique capability of our method, opening up a new direction towards more user-friendly video generation.

C.3. Scene Cut Conditioning

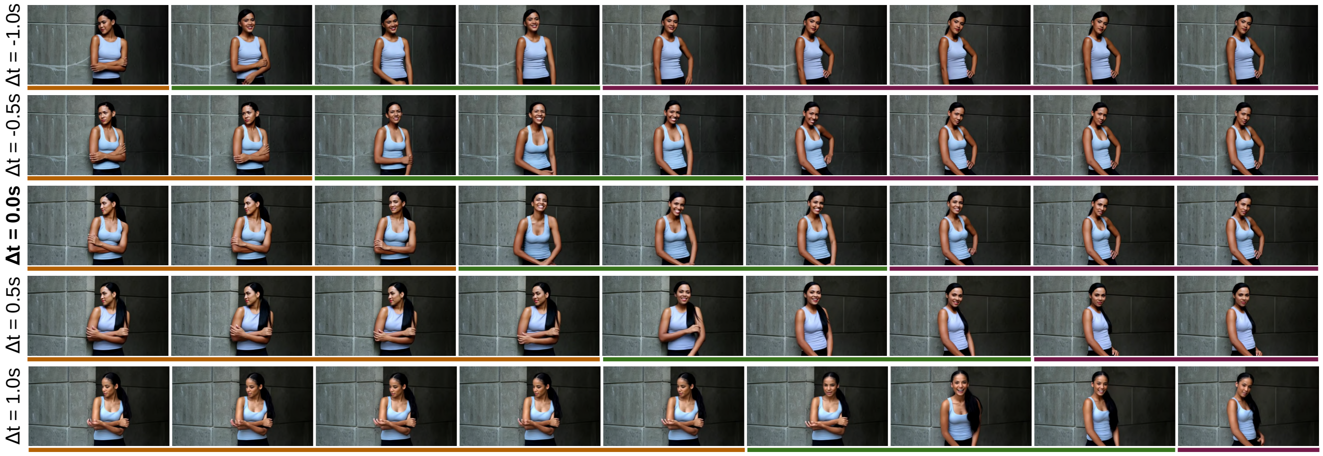
As shown in the ablation, removing scene cut conditioning leads to undesired shot transitions in generated videos. A closer inspection reveals that the generation of cuts is sensitive to the text prompt of an event. When it contains a description of the camera shot (e.g., "a close-up view of"), it is more likely to introduce a cut. In contrast, explicitly conditioning on scene cuts frees us from this issue.

We show some qualitative scene cut control results in Fig. 14. MinT is able to generate shot transitions at desired timestamps, while preserving subject identities. When zeroing out the scene cut input, we can get cut-free videos which validates our design. Finally, we show that our model can switch between sudden camera changes or gradual zoom-in and zoom-out effects, enabling fine-grained control.

An interesting direction is to learn different types of scene transitions such as jump cut, dissolve, and wipe. Since our goal is to retain training data instead of learning fancy transition control, we leave this for future work.

C.4. Event Time Span Control

MinT supports fine-grained control of event time span. To show this, we take a sample from our dataset and offset the start and end timestamps of all events by a specific value. Fig. 15 presents the results, where each video generates events following its new timing. In addition, we can roughly keep the appearance of the main subject and background unchanged. MinT is the first video generator in the literature that achieves this control ability. We view it as an important step towards a practical content generation tool.



Prompts: *"A woman stands with head turns left and arms crossed"* → *"looks at the camera, puts her hands down and laughs"* → *"puts her left hand resting at waist level"*

Figure 15. **Generated videos with different event time spans.** In each example, we offset the start and end timestamps of all events by a specific number of seconds. Results show that MinT enables fine-grained event timing control while keeping the subjects' appearances to be roughly the same. This capability is very useful for controllable video generation. Please see our project page for more results.

Method	FID ↓	FVD ↓	CLIP-score ↑
Task: T2V (a.k.a. story generation in [12])			
Phenaki	273.41	998.19	0.210
Ours	**40.87**	**484.44**	**0.284**
Task: I2V (a.k.a. story continuation in [12])			
Phenaki	240.21	674.5	0.219
Ours	**21.85**	**314.59**	**0.273**

Table 4. **Comparison with Phenaki on StoryBench.** We compare with the zero-shot variant Phenaki-Gen-ZS in their paper [12] since our model is not fine-tuned on StoryBench. We clearly outperform Phenaki across all metrics in both tasks.

C.5. StoryBench Comparison with Phenaki

The original StoryBench paper [12] proposed a baseline for their dataset, which runs Phenaki [87] to generate events in an autoregressive way. However, they conducted evaluation on a much lower resolution (160×96), and neither their code nor pre-trained weights were released, making a direct comparison hard. We still compare with them in Tab. 4 for completeness. We only report metrics that both papers evaluate, which cover visual quality (FID, FVD) and text alignment (CLIP-score). MinT significantly outperforms Phenaki across all metrics in both T2V and I2V tasks. This demonstrates the effectiveness of fine-tuning from a large-scale pre-trained video model.

C.6. Comparison with SOTA Video Generators

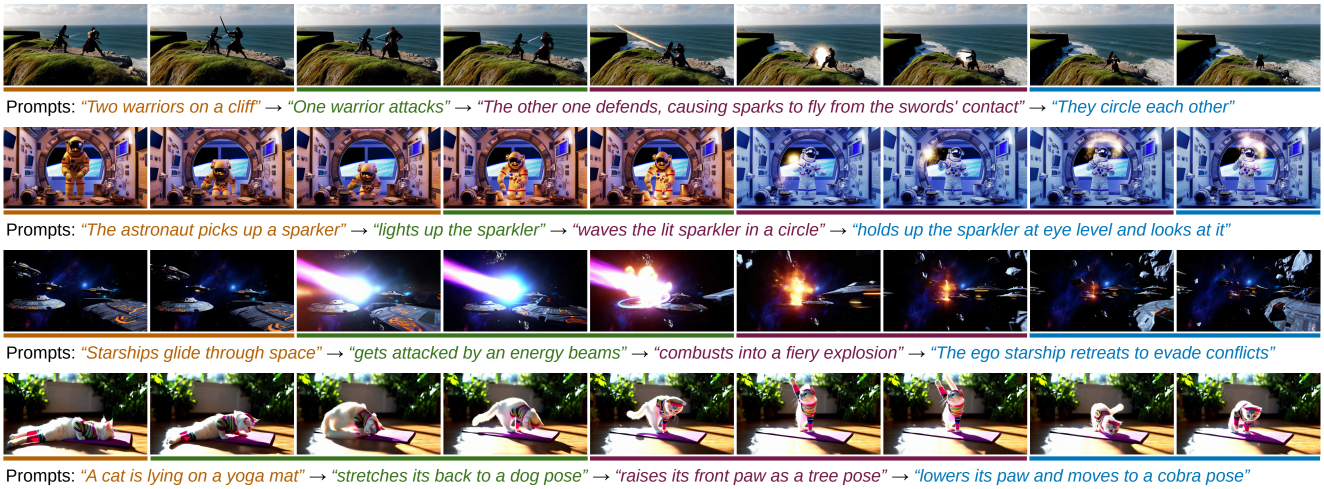
To show that sequential event generation is a common failure case of even SOTA video generators, we present more results in Fig. 17 and Fig. 18. One surprising observation we had is that, when using prompts following the official guideline of these models (e.g., using the LLM provided by CogVideoX to enhance prompts), the model only generates the first event and ignores all subsequent ones. Only if we directly concatenate event captions without specifying global properties such as camera motion, background description, and detailed subject attributes (i.e., directly use prompts like "A person first do A, then do B, and finally do C"), the model starts to generate some events transitions.[3] One possible cause is that in the training data of these models, videos with sequential events are never annotated with such detailed global properties. However, since we do not have access to their training details, we can't figure out the true reason behind it. Therefore, we just use naively concatenated prompts to generate all results. The prompts we used for these models can be found on our project page. Notably, this workaround prevents us from using detailed captions to control the scene and subjects, which greatly affects the controllability of these models.

Still, when prompted with a text that contains multiple events, these models have three common failure modes:

1. Only generates partial events, and completely ignores the remaining ones. For example, in the third example in Fig. 17, all models miss the "blow kiss" action;
2. Generates events in the wrong order or "merge" multiple events. For example, in the last example in Fig. 17, Kling 1.5 generates the man with his hand under his mouth at the beginning of the video. Yet, this should happen last;
3. Bind wrong actions or properties to subjects. For example, in the first example in Fig. 18, Gen-3 Alpha generates a woman coming into the frame instead of a man.

Remark. There might be other ways to fix this issue without using temporally-grounded captions as in MinT. For example, one may fine-tune the model on video datasets annotated with detailed sequential event information [97]. Still, this will not allow precise control over the start and end times of events, which is a unique capability of our model.

[3] The detailed prompt does not exceed the maximum input text length of these models, so context length is not the reason here.



Prompts: *"Two warriors on a cliff"* → *"One warrior attacks"* → *"The other one defends, causing sparks to fly from the swords' contact"* → *"They circle each other"*

Prompts: *"The astronaut picks up a sparker"* → *"lights up the sparkler"* → *"waves the lit sparkler in a circle"* → *"holds up the sparkler at eye level and looks at it"*

Prompts: *"Starships glide through space"* → *"gets attacked by an energy beams"* → *"combusts into a fiery explosion"* → *"The ego starship retreats to evade conflicts"*

Prompts: *"A cat is lying on a yoga mat"* → *"stretches its back to a dog pose"* → *"raises its front paw as a tree pose"* → *"lowers its paw and moves to a cobra pose"*

Figure 16. **Generated videos with out-of-distribution prompts.** After fine-tuning, MinT still possesses the base model's ability to generate novel concepts. Please see our project page for more results.

C.7. Out-of-Distribution Prompts

MinT is fine-tuned on temporal caption videos that mostly describe human-centric events. In the paper, we have shown some non-human results such as animals and traffics. Here, we show that our model still possesses the ability to generate novel concepts and their combinations, which is an important property of large-scale pre-trained video generators. As shown in Fig. 16, MinT generates out-of-distribution characters such as warriors and astronaut, scenes such as starships in the space, and non-existing events such as a cat doing yoga. This proves that our model does not forget the rich pre-training knowledge in the base model.

D. Limitations and Future Works

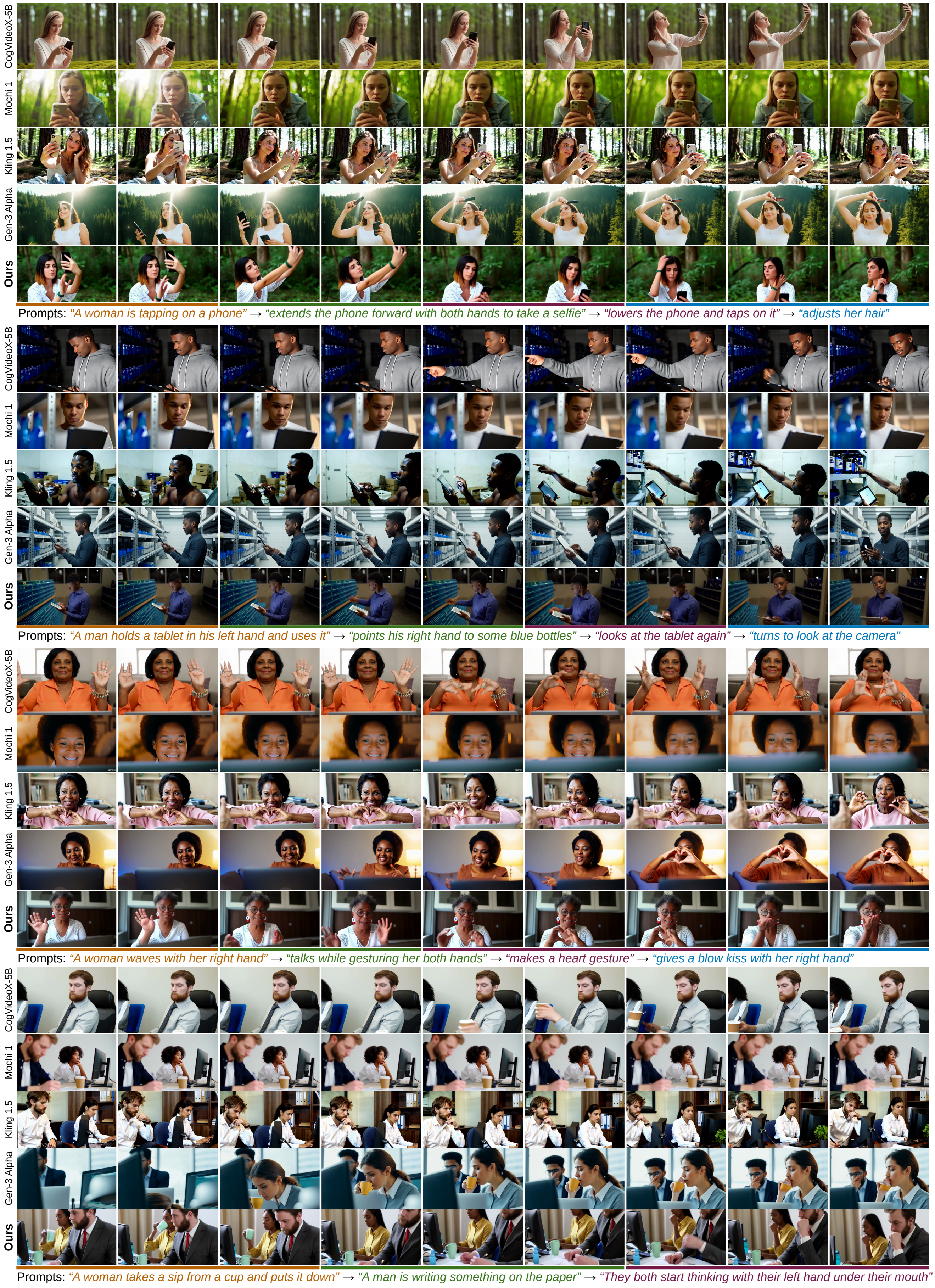
MinT is fine-tuned from a pre-trained text-to-video diffusion model, and thus we are bounded by the capacity of the base model. For example, it is challenging to generate human hands or scenes involving complex physics.

When generating an event involving multiple subjects, MinT may fail to associate attributes and actions to the correct subject. Similar to the *temporal binding* problem we try to address in this paper, we believe this issue can be solved with *spatial binding*. For example, by grounding subjects with bounding boxes and attribute labels [47, 49, 90].

Finally, MinT sometimes fails to associate entities specified in the global caption and temporal captions. Such association requires complex reasoning of the text conditioning, and may be resolved by simply scaling up the training data.

Please refer to our project page for video examples and detailed analysis of these failure cases.

Future works. It is interesting to enhance our model with recent progress in training-free long video generation techniques [31, 70, 88]. Another direction is to combine MinT with video personalization methods [40, 50, 54] to enable both fine-grained control within a shot and subject consistency across shots for minute-long video creation.



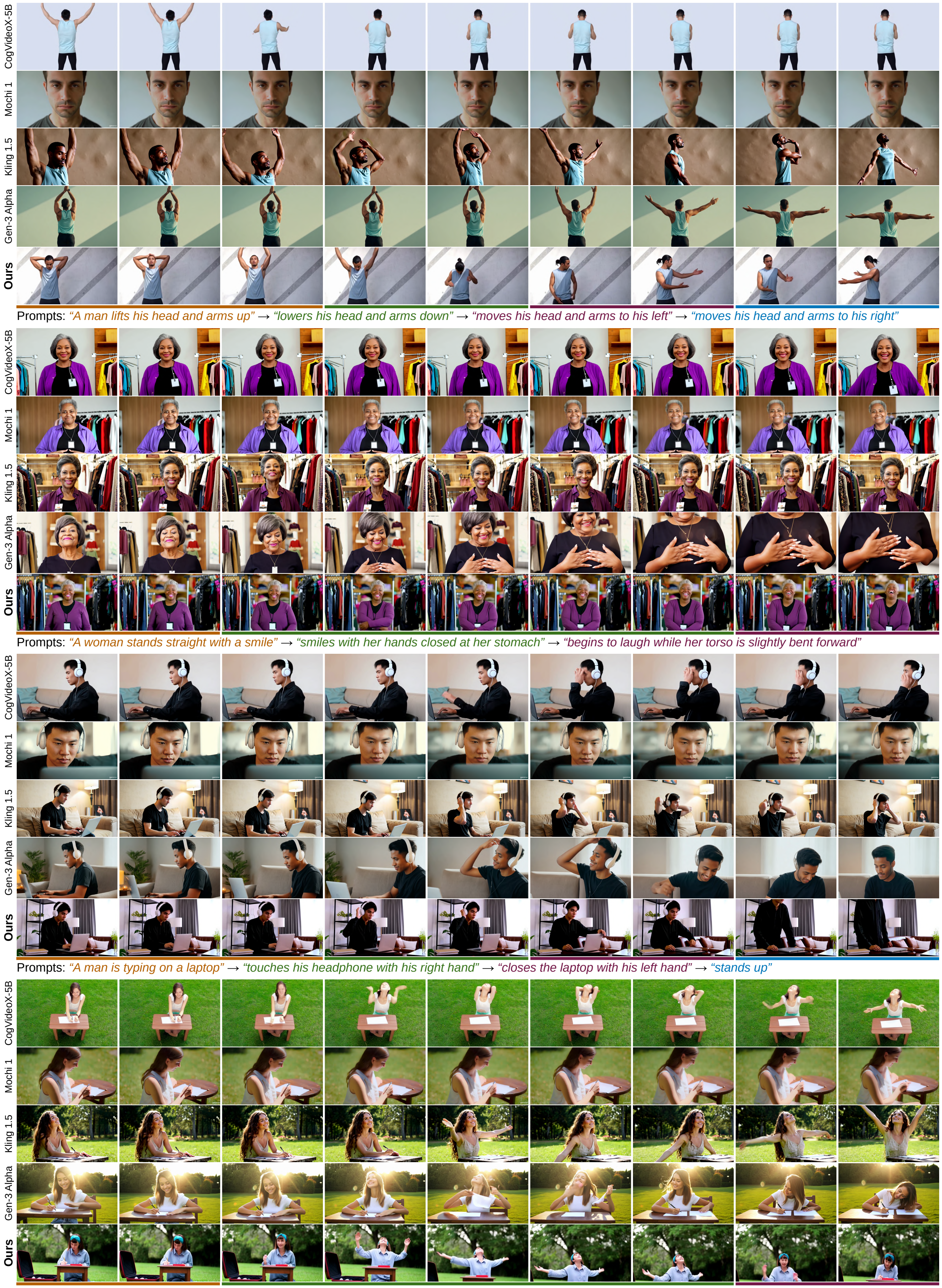
Prompts: *"A woman is tapping on a phone"* → *"extends the phone forward with both hands to take a selfie"* → *"lowers the phone and taps on it"* → *"adjusts her hair"*

Prompts: *"A man holds a tablet in his left hand and uses it"* → *"points his right hand to some blue bottles"* → *"looks at the tablet again"* → *"turns to look at the camera"*

Prompts: *"A woman waves with her right hand"* → *"talks while gesturing her both hands"* → *"makes a heart gesture"* → *"gives a blow kiss with her right hand"*

Prompts: *"A woman takes a sip from a cup and puts it down"* → *"A man is writing something on the paper"* → *"They both start thinking with their left hand under their mouth"*

Figure 17. **More comparisons with SOTA video generators.** We run SOTA open-source models CogVideoX [100] and Mochi [82], and commercial models Kling 1.5 [2] and Gen-3 Alpha [1] using their online APIs. Please see our project page for video results.



Prompts: *"A man lifts his head and arms up"* → *"lowers his head and arms down"* → *"moves his head and arms to his left"* → *"moves his head and arms to his right"*

Prompts: *"A woman stands straight with a smile"* → *"smiles with her hands closed at her stomach"* → *"begins to laugh while her torso is slightly bent forward"*

Prompts: *"A man is typing on a laptop"* → *"touches his headphone with his right hand"* → *"closes the laptop with his left hand"* → *"stands up"*

Prompts: *"A woman is writing something on a table"* → *"looks upwards with a smile and spreads her arms"* → *"resumes to write something on the table"*

Figure 18. **More comparisons with SOTA video generators.** We run SOTA open-source models CogVideoX [100] and Mochi [82], and commercial models Kling 1.5 [2] and Gen-3 Alpha [1] using their online APIs. Please see our project page for video results.