TIE: TIME INTERVAL ENCODING FOR VIDEO GENERATION OVER EVENTS

Anonymous authors
Paper under double-blind review

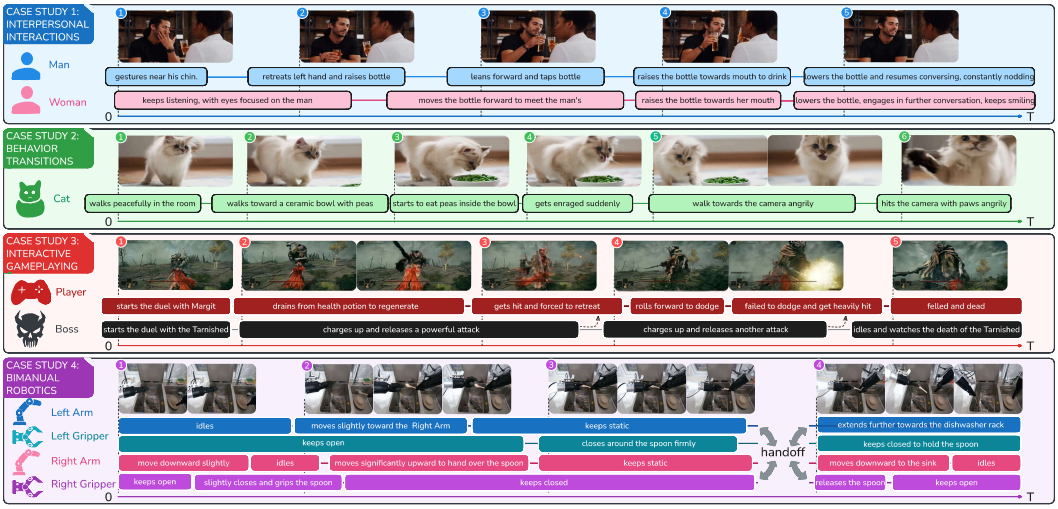

Figure 1: Generation results with **TIE**. By leveraging **Time Interval Encoding (TIE)**, the model generates videos from structured event descriptions with explicit temporal boundaries. The examples demonstrate accurate event alignment under **concurrent** and **overlapping** event settings beyond standard sequential prompt schedules, including multi-entity dynamics and interleaved interactions.

ABSTRACT

Director-style prompting, robotic action prediction, and interactive video agents require temporal grounding over concurrent events. Point-wise positional encodings do not explicitly represent the temporal support of event intervals, while standard sequential prompt schedules do not directly specify overlapping events. We propose **Time Interval Encoding (TIE)**, an interval-aware generalization of rotary embeddings for DiT cross-attention. TIE integrates positional evidence over an event's temporal support and calibrates its matching strength. Under a uniform kernel, it yields a closed-form sinc-modulated rotary encoding, RoTE, that retains the standard attention interface and reduces to RoPE in the point-wise limit. Interval integration also provides a boundary-noise sensitivity bound under non-degenerate normalization. On *OmniEvents*, TIE improves human-verified Temporal Constraint Satisfaction Rate from 77.34% to 96.03% and reduces temporal boundary error from 0.261s to 0.073s, while maintaining comparable visual quality on the primary DiT backbone. Trajectory-level metrics, noisy-timestamp experiments, and learning curves support improved temporal alignment and faster convergence. Gameplay and robotics examples illustrate control over interacting events and edited onset times. We will open-source all model weights, code, and data.

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

1 INTRODUCTION

Diffusion Transformers (DiT) have improved video fidelity (Peebles & Xie, 2023; Wan et al., 2025; Lu et al., 2024; Ma et al., 2025; Yang et al., 2025; Singer et al., 2023; Blattmann et al., 2023) and creative control (Chen et al., 2023; Jiang et al., 2025; Wang et al., 2023b; Guo et al., 2024; Yang et al., 2024b). They are also being studied as world simulators (Hu et al., 2023), robotic data generators (Agarwal et al., 2025; Zhou et al., 2024; Huang et al., 2025; Zhu et al., 2025), and interactive environments (Yang et al., 2024a; Bruce et al., 2024; Du et al., 2023; Valevski et al., 2025). These applications require precise event boundaries, ordering, and overlap: for example, coordinating two robotic arms or multiple athletes (Li et al., 2021).

One bottleneck in achieving such precision is a **representation deficit** in current temporal positional encodings. While videos are continuous spatiotemporal signals, modern DiTs predominantly treat time as a sequence of discrete *points*. Standard rotary positional encodings (**RoPE**), originally designed for text, are point-wise by nature (Su et al., 2024). This formulation leaves interval support implicit: while real-world events (e.g., "a robot arm gripping a tool") occupy continuous **time intervals**, the model processes them as if they were tied to infinitesimal timestamps. Prompt concatenation can leave event timing ambiguous; scaling alone does not explicitly add interval support. Moreover, in large-scale real-world video corpora, event intervals are rarely available with frame-perfect boundaries: they are often produced by VLMs, action detectors, or other automatic annotators, whose start and end times are inevitably noisy. Thus, a useful interval representation must not only encode temporal support, but also remain stable under imperfect boundary annotations.

Existing works on timeline-guided synthesis (Villegas et al., 2023; Oh et al., 2024; Wu et al., 2025; Lin et al., 2024; Qiu et al., 2024; Cai et al., 2025; Yin et al., 2023) often assume a "single-active-prompt" constraint, where only one text prompt is valid at any given timestamp. This is insufficient for complex physical interactions where events are frequently **interleaved or staggered**. For instance, in a bimanual assembly task, the "reaching" phase of one arm must be precisely coordinated with the "holding" phase of the other. To bridge this gap, video generation models require an encoding mechanism that elevates time intervals to first-class primitives, allowing for a principled handling of event duration and concurrency.

This paper introduces **Time Interval Encoding (TIE)**, a novel interval-aware formulation that explicitly models event durations and boundaries within the cross-attention mechanism. Our starting point is two basic principles: *Temporal Integrability*, which requires an event's contribution to aggregate positional evidence over its full duration, and *Duration Invariance*, which calibrates expected self-overlap at the interval center. We show that, within RoPE-compatible bilinear attention, kernel integration and an explicit self-overlap calibration determine the encoder for a fixed kernel. Under the uniform kernel, these choices yield a closed-form sinc-modulated encoder (**RoTE**) that retains the standard attention interface and asymptotic attention complexity and reduces to **RoPE** in the point-wise limit $r \to 0$. The same interval-integrated form also acts as a temporal low-pass filter: boundary perturbations affect only the marginal portion of the event support, yielding a sensitivity bound under non-degenerate normalization.

We construct *OmniEvents*: 250k general-video PexelsEvents clips, 86k RoboticsEvents clips, and 80k GameEvents clips from Elden Ring. Using Wan2.2 TI2V-5B, we evaluate visual quality, interval controllability, noisy-boundary robustness, and the contribution of interval integration. TIE improves human-verified TCSR from 77.34% to 96.03% and temporal error from 0.261s to 0.073s, improves trajectory-level alignment, and reaches comparable quality in fewer training steps. Gameplay examples illustrate localized editing of later interactions.

Our contributions are summarized as follows:

- We propose Time Interval Encoding (**TIE**) for explicit concurrent-event conditioning. Uniform-kernel integration and self-overlap calibration yield the sinc-modulated encoder (**RoTE**), retaining the standard QK attention interface.
- We establish a structured-prompt evaluation setting for interval-conditioned video generation, including the *OmniEvents* dataset and human-verified temporal-grounding metrics such as TCSR and trajectory-level metrics.

- We bound sensitivity to bounded temporal perturbations under non-degenerate normalization and evaluate noisy boundaries, a core requirement for large-scale event-conditioned pretraining where interval annotations are often produced by imperfect automatic annotators.
- We show empirically that **TIE** maintains comparable visual quality on the primary backbone, substantially improves human-verified and trajectory-level temporal grounding, accelerates the convergence of temporal grounding, and supports controllable multi-subject interactions.

2 Related Work

Multi-event video generation conditions the synthesis on a sequence of event-specific prompts. Common sequential evaluation protocols use a single-active-prompt constraint: at most one event prompt is valid at any given timestamp. Sequential-token approaches such as Phenaki (Villegas et al., 2023) translate disjoint prompts into a continuous token stream; diffusion-based extenders Gen-L-Video (Wang et al., 2023a) and MEVG (Oh et al., 2024) denoise non-overlapping local windows; multi-agent systems DreamFactory (Xie et al., 2024) and VGoT (Zheng et al., 2024) structure transitions between sequentially scheduled events. Related temporal-conditioning approaches include the following: MinT (Wu et al., 2025) aligns multi-event features through positional embeddings indexed at scalar timestamps, TS-Attn (Zhang et al., 2026) applies separable mask-based attention over disjoint prompt-time slots, and LongLive (Yang et al., 2026) preserves single-prompt KV state via KV-recache. These sequential protocols do not directly evaluate overlapping prompts; this does not establish a mathematical inability to represent concurrency. Yet physically realistic interactions—bimanual manipulation, multi-subject sports, adversarial combat—are inherently concurrent: in *OmniEvents*, 68% of general-video events and over 99% of robotics and gameplay events overlap with another event. Closing this gap requires elevating the time interval itself to a first-class primitive, rather than retrofitting concurrency onto sequential schedules.

3 Method

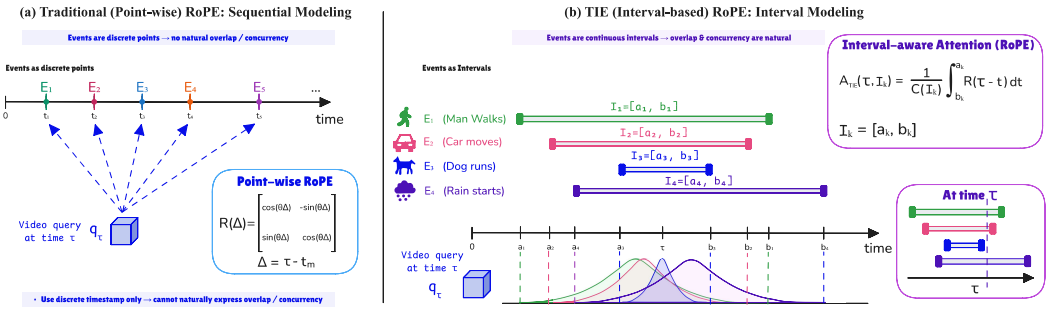

Figure 2: Comparison between normal RoPE (Point-wise) and **TIE** (Interval-based): The point-wise RoPE cannot naturally model the point-to-interval event activation. While our **TIE** supports natural modeling of time interval.

3.1 Time Interval Encoding

To bridge point-wise video features and interval-based textual events, we propose **Time Interval Encoding (TIE)**. We treat a video token as a temporal point with midpoint timestamp $m_i \in \mathbb{R}$, and a textual event token as a time interval $I_j = [t_j^s, t_j^e]$ (find visualization explanation in 2). The goal is to extend **RoPE** from point-to-point matching to point-to-interval matching while preserving the standard dot-product attention interface.

RoPE Review. For a token $\mathbf{z} \in \mathbb{R}^d$ at time t, standard **RoPE** applies $\mathrm{RoPE}(\mathbf{z}, t) = \mathbf{R}_t \mathbf{z}$, where

$$\mathbf{R}_t = \mathrm{diag}(\mathbf{A}_{1,t}, \mathbf{A}_{2,t}, \ldots, \mathbf{A}_{d/2,t}), \qquad \mathbf{A}_{i,t} = \begin{bmatrix} \cos(\theta_i t) & -\sin(\theta_i t) \\ \sin(\theta_i t) & \cos(\theta_i t) \end{bmatrix}. \tag{1}$$

Its key property is

$$s_{\text{rope}}(\mathbf{q}, \mathbf{k}; t_q, t_k) = \text{RoPE}(\mathbf{q}, t_q)^\top \text{RoPE}(\mathbf{k}, t_k) = \mathbf{q}^\top \mathbf{R}_{t_k - t_q} \mathbf{k}. \tag{2}$$

RoPE therefore models relations between temporal *points*; **TIE** extends it to temporal *intervals*.

Core TIE Formulation. We define a point-to-interval score using the following two principles.

Definition 3.1 (Two Principles of **TIE**)**.** An interval-aware score satisfies the **TIE** principles if:

- (*Temporal Integrability.*) The raw interval score integrates point-wise **RoPE** evidence according to a chosen probability kernel on the interval. Formally, for some probability kernel μ_I supported on I with $\mu_I(I) = 1$,
$$\bar{s}(\mathbf{q}, \mathbf{k}; t_q, I) := \mathbb{E}_{\tau \sim \mu_I}[s_{\text{rope}}(\mathbf{q}, \mathbf{k}; t_q, \tau)].$$
- (*Duration Invariance.*) The final score compensates for averaging attenuation through the self-overlap calibration below, without making arbitrary scores invariant to duration. Accordingly,
$$s(\mathbf{q}, \mathbf{k}; t_q, I) := \frac{1}{C(\mu_I)} \bar{s}(\mathbf{q}, \mathbf{k}; t_q, I), \qquad C(\mu_I) > 0,$$
where $C(\mu_I)$ depends only on μ_I and is independent of $\mathbf{q}$, $\mathbf{k}$, and t_q.

Within bilinear attention, integration fixes the linear encoder up to its scalar calibration. First, by *Temporal Integrability*,

$$\bar{s} = \mathbb{E}_{\tau \sim \mu_I}[s_{\text{rope}}(\mathbf{q}, \mathbf{k}; t_q, \tau)] = \text{RoPE}(\mathbf{q}, t_q)^\top \mathbb{E}_{\tau \sim \mu_I}[\text{RoPE}(\mathbf{k}; \tau)]. \tag{3}$$

Therefore, **TIE** must be the form:

$$\text{TIE}(k, I) \propto \mathbb{E}_{\tau \sim \mu_I}[\text{RoPE}(\mathbf{k}; \tau)].$$

We additionally fix the coefficient by requiring unit expected self-overlap at the center:

$$1 = \mathbb{E}_{\mathbf{q} \sim \mathcal{U}(S^{d-1})}[\text{RoPE}(\mathbf{q}, c)^\top \text{TIE}(\mathbf{q}, I)],$$

where c is the interval midpoint. The unit-sphere distribution is a calibration reference, not an assumption about learned queries. Solving the equation above shows that the coefficient does not depend on a specific $\mathbf{q}$:

$$\text{TIE}(\mathbf{k}, I) = \frac{1}{C(\mu_I)} \mathbb{E}_{\tau \sim \mu_I}[\text{RoPE}(\mathbf{k}, \tau)]. \tag{4}$$

The corresponding score will be

$$s(\mathbf{q}, \mathbf{k}, t_q, I) := \text{RoPE}(\mathbf{q}, t_q)^\top \text{TIE}(\mathbf{k}, I). \tag{5}$$

3.2 A Closed-Form Instantiation: Uniform Kernel Yields Sinc-Modulated RoPE

Fixing the bilinear attention form, RoPE frequencies, uniform probability kernel, and explicit self-overlap calibration gives the following closed form.

Theorem 3.2 (Uniform-kernel **TIE** yields the closed-form **RoTE**)**.** *Given $I = [t^s, t^e]$, let $\mu_I = \mathcal{U}([t^s, t^e])$, and define the interval center and radius by $c = (t^s + t^e)/2$ and $r = (t^e - t^s)/2$. Assuming $C_r > 0$ and the stated self-overlap calibration, the encoder has the closed form*

$$\mathbf{A}_{i,c,r} = \text{sinc}(\theta_i r) \begin{bmatrix} \cos(\theta_i c) & -\sin(\theta_i c) \\ \sin(\theta_i c) & \cos(\theta_i c) \end{bmatrix}. \tag{6}$$

Stacking the blocks gives $\boldsymbol{R}_{c,r} = \text{diag}(\mathbf{A}_{1,c,r}, \mathbf{A}_{2,c,r}, \ldots, \mathbf{A}_{d/2,c,r})$, and the resulting interval encoder is $\text{RoTE}(\boldsymbol{k}, c, r) = \frac{1}{C_r} \boldsymbol{R}_{c,r} \boldsymbol{k}$, where $C_r = \frac{2}{d} \sum_{\ell=1}^{d/2} \text{sinc}(\theta_\ell r)$. Consequently, for a query token $\boldsymbol{q}_i$ at time m_i and a key token $\boldsymbol{k}_j$ associated with interval $I_j = [t_j^s, t_j^e]$, the attention score becomes

$$\boldsymbol{s}_{i,j} \propto \text{RoPE}(\boldsymbol{q}_i, m_i)^\top \text{RoTE}(\boldsymbol{k}_j, c_j, r_j), \qquad c_j = \frac{t_j^s + t_j^e}{2},\ r_j = \frac{t_j^e - t_j^s}{2}. \tag{7}$$

The normalization constant is chosen to preserve unit expected self-overlap at the interval center, compensating for attenuation from averaging rotations. The center c controls the rotation phase, while the radius r suppresses high-frequency bands through $\text{sinc}(\theta r)$. Long intervals therefore act as a temporal low-pass filter, and **RoTE** reduces to standard **RoPE** as $r \to 0$. This solution is specific to the fixed kernel and calibration; the two principles alone do not uniquely select either.

Boundary-only Ablation. For ablations we also consider the channel-wise, endpoint-calibrated Dirac-kernel variant **DoTE**, which keeps only the interval boundaries: $\mathrm{DoTE}(\mathbf{k}, I) = \mathbf{Concat}(\mathrm{RoPE}(\mathbf{k}^{(s)}, t^s), \mathrm{RoPE}(\mathbf{k}^{(e)}, t^e))$. Unlike **RoTE**, this variant ignores the interval interior. Its formal derivation is also deferred to Section A.1. In practice, we scale every timestamp with a scaling factor γ, that is, $c = \gamma\frac{t^s+t^e}{2}, r = \gamma\frac{t^e-t^s}{2}$, so that we can better control the decaying speed. By default we choose $\gamma = 4.0$.

3.3 **RoTE** FOR LARGE-SCALE PRETRAINING

At scale, event intervals are often produced by VLMs, action detectors, or other automatic annotators with imperfect start and end times. RoTE integrates positional evidence over the interval instead of encoding only its center or boundaries. For endpoint errors bounded by $\delta < r$, Theorem A.4 gives an $O(\|\mathbf{q}\|\|\mathbf{k}\|\delta/r)$ sensitivity bound when the normalization stays bounded away from zero. Appendix A.1 retains the complete theorem, proof, low-pass interpretation, and implications for learned embeddings and large-scale pretraining.

4 EXPERIMENTS

We evaluate TIE on visual quality, concurrent-event controllability, convergence, and robustness to noisy temporal annotations. RoTE is the default uniform-kernel implementation. Sequential methods including MinT (Wu et al., 2025), TS-Attn (Zhang et al., 2026), and MEVG (Oh et al., 2024) motivate complementary evaluation on native sequential benchmarks (Section 4.3 and Appendix D).

4.1 EXPERIMENTAL SETUP

Backbone. We use Wan2.2-5B-TI2V as the base video generator, following its standard DiT architecture and training recipe (Wan et al., 2025). **TIE** is inserted only into the text-to-video cross-attention pathway by replacing point-wise temporal **RoPE** on event tokens with interval-aware key encodings.

Datasets. We construct the *OmniEvents* dataset, which consists of three complementary domains. For general event-conditioned video generation, we collect 250k clips from Pexels-400k (Zhi-Min, 2023), denoted as *PexelsEvents*. For high-precision action-centric evaluation, we collect 80k clips of EldenRing gameplay videos with frame-accurate event intervals recorded from game memory, denoted as *GameEvents*. For task-specific domain applications, we annotate 86k clips from the Agibot-World-Alpha (contributors, 2024), forming the *RoboticsEvents* part. Each training sample is represented as a set of event tokens, where each event is associated with a textual description and a time interval (Figure A4 shows an example case). Details about the *OmniEvents* can be found in Appendix B.2.

Baselines. We compare against several variants: **Base**, the original Wan2.2 model without event-interval fine-tuning; **Finetuned**, a standard event-conditioned fine-tuning baseline that concatenates event descriptions and timestamps into the prompt but keeps point-wise temporal encoding; **DoTE**, a boundary-only variant that encodes only the start and end timestamps; and **RoTE**, our main method. The original ablations compare timestamp prompting, boundary-only encoding, and interval integration. Additional interval baselines and measured cost appear in Appendix F; bias/mask efficiency depends on implementation.

Evaluation metrics. We evaluate three aspects. First, for visual quality and semantic alignment, we report FID, FVD, and VideoScore dimensions including visual quality, temporal consistency, dynamic degree, text alignment, and factual consistency (Heusel et al., 2017; Unterthiner et al., 2018; He et al., 2024). These metrics test whether **TIE** preserves the base model's visual generation ability. Second, for time interval controllability, we use both human-verified event-level metrics and temporal alignment metrics. Our primary human metric is Temporal Constraint Satisfaction Rate (TCSR), which summarizes the total temporal alignment degree, together with breakdowns into Event Occurrence, Temporal Error, Order Accuracy and Overlap Accuracy. We report CLIP-Event, EMD and nDTW as trajectory-level temporal alignment proxies. CLIP-Event calculates the per-event CLIP

Score for generated videos at frame-level, while EMD and nDTW compare the generated videos with reference ones in temporal alignment. These metrics complement TCSR by measuring whether the generated visual or semantic trajectory follows the target temporal structure.

4.2 **TIE** PRESERVES VISUAL QUALITY AND DIT COMPATIBILITY

A temporal-control module should retain the visual prior of its pretrained backbone. Table 1 reports FID, FVD, and VideoScore on PexelsEvents. Relative to Finetuned, TIE obtains comparable or better FID/FVD and similar visual quality, temporal consistency, text alignment, and factual consistency. These results support its compatibility with the primary DiT backbone while improving temporal grounding.

Table 1: Quality on PexelsEvents. I2V uses the first frame as a visual condition. VQ/TC/DD/TA/FC denote visual quality, temporal consistency, dynamic degree, text alignment, and factual consistency.

Text-to-video (T2V)							
Method	FID↓	FVD↓	VQ↑	TC↑	DD↑	TA↑	FC↑
Base	59.68	357.51	2.73	2.78	2.58	2.68	2.65
Finetuned	43.74	234.40	3.03	3.01	2.97	2.86	2.94
RoTE	**42.53**	**217.29**	**3.10**	**3.05**	**3.06**	**2.92**	**2.99**
Image-to-video (I2V)							
Method			VQ↑	TC↑	DD↑	TA↑	FC↑
Base			2.98	2.94	2.93	2.87	2.88
Finetuned			3.26	3.21	**3.28**	3.06	3.12
RoTE			**3.30**	**3.22**	3.27	**3.07**	**3.18**

4.3 **TIE** IMPROVES TIME INTERVAL CONTROLLABILITY

We next evaluate the central claim of this work: whether **TIE** improves the model's ability to generate events at specified time intervals, including sequential, overlapping, and concurrent event structures.

Human-Verified Temporal Constraint Satisfaction. Standard visual-quality metrics may assign similar scores to videos whose actions occur at different times. We therefore use a structured human verification protocol. We sample 100 prompts and generate 100 Finetuned videos and 100 TIE videos. Ten annotators, blinded to model identity, verify event occurrence and record start/end deviations from each requested interval $[t_s, t_e]$. They answer event-verification questions rather than open-ended preference questions; metrics are computed from aggregated annotations.

We report five metrics: **Event Occurrence** measures event realization; **Temporal Error** measures boundary deviation with duration-based missing-event penalties; **Order Accuracy** measures before/after consistency; **Overlap Accuracy** measures concurrency preservation; and **Temporal Constraint Satisfaction Rate (TCSR)** is the fraction of satisfied event and relation constraints. Each start and end contributes a separate constraint, satisfied within 0.25s of its target if the event occurs; missing events fail both. Order and overlap constraints require both events to occur and preserve the requested relation. Appendix C gives the full protocol and formulas.

Figure 3(a) shows improvements across all five human metrics. Figure 4 illustrates more accurate event onset and multi-event responses than prompt concatenation alone, complementing the aggregate temporal-control results.

Trajectory-Level Temporal Alignment. We complement human-verified TCSR with nDTW (Sakoe & Chiba, 1978), EMD, and CLIP-Event. nDTW reports a higher-is-better score from temporal warping alignment; EMD measures temporal distribution distance (lower is better); CLIP-Event averages frame-level semantic alignment to event descriptions. RoTE improves all three in Figure 3(b), supporting the event-level results with trajectory-level evidence.

Comparison on Baselines' Native Regime. StoryEval (Wang et al., 2024) and StoryBench (Bugliarello et al., 2023) complement concurrent-event evaluation with sequential protocols de-

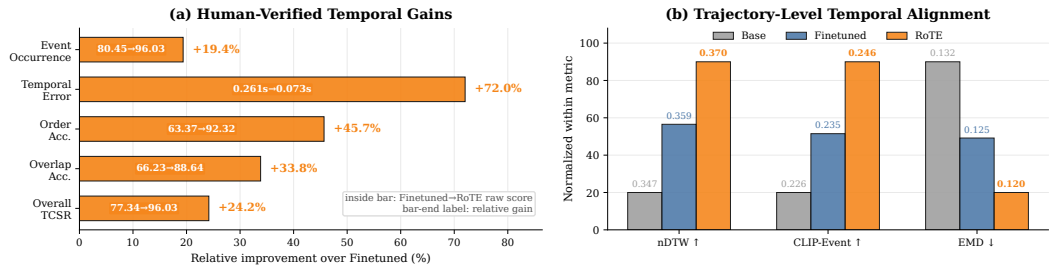

Figure 3: **Temporal alignment improvements of TIE.** (a) Human-verified metrics show that RoTE consistently improves over finetuning across event occurrence, temporal error, ordering accuracy, overlap accuracy, and overall TCSR. Each bar shows the relative improvement over the finetuned baseline, while the in-bar label reports the raw score change from finetuning to **TIE**. (b) Trajectory-level metrics further confirm the improvement: **TIE** achieves higher nDTW and CLIP-Event scores and lower EMD, indicating better temporal alignment at the trajectory level.

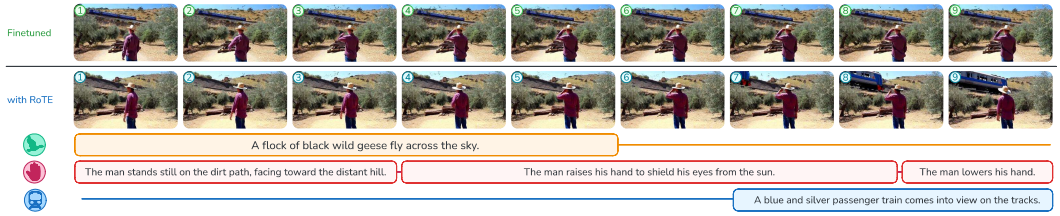

Figure 4: Text-to-video generation example. The **Finetuned** baseline misses some requested events (e.g., "the man puts his hand off") and misaligns event onset (e.g., the train remains static in the frame instead of entering at 7.0s), whereas the **TIE**-enhanced variant responds at the correct time.

signed for prior methods. StoryEval prompts differ from our training distribution, and its 5s condition differs from our mainly 10s training clips. Table 2 separates these lengths: Finetuned does not improve the 5s average over Base, but improves the 10s average from 23.9% to 38.8%. RoTE improves over Finetuned to 49.4% at 5s and 52.3% at 10s. Appendix D provides the original StoryBench and prompt-extension results.

Faster Convergence of Temporal Grounding. Beyond final accuracy, we evaluate whether **TIE** accelerates the learning of temporal grounding. We train Finetuned and **TIE** variants under identical data (*RoboticsEvents*), batch size, learning rate, and training schedule, and measure temporal metrics at matched training steps. As shown in Figure 5, TIE reaches the same temporal-control level (nDTW, EMD) and visual quality (FVD, CLIP-Video) (it is noted that CLIP-Video is different from CLIP-Event) in substantially fewer steps. This suggests that interval-aware encoding improves the efficiency of learning temporal grounding, rather than merely adding capacity or overfitting to the final evaluation metrics.

Table 2: StoryEval evaluation results under 10s and 5s generation settings. The 10s setting uses 161 frames at 16 fps; the 5s setting uses 121 frames at 24 fps. Results at different durations are separate comparisons. All methods are evaluated at 720P. TS-Attn is reproduced at 720P rather than its original 480P setting.

Method	Human	Animal	Object	Retrieval	Creative	Easy	Hard	Average
Base (10s)	27.6%	27.2%	20.3%	43.1%	13.9%	34.5%	5.9%	23.9%
Finetuned (10s)	39.0%	41.0%	35.7%	53.1%	25.3%	55.5%	21.5%	38.8%
RoTE (10s)	58.5%	52.3%	48.4%	53.8%	42.7%	64.4%	38.8%	52.3%
Base (5s)	41.1%	41.7%	35.9%	51.2%	28.0%	58.6%	20.2%	39.7%
TS-Attn(Zhang et al., 2026) (5s)	42.2%	45.4%	36.6%	52.8%	33.0%	56.9%	25.5%	41.8%
Finetuned (5s)	41.9%	39.8%	37.1%	50.8%	26.7%	55.8%	23.0%	39.3%
RoTE (5s)	54.7%	52.8%	42.4%	53.1%	36.3%	61.2%	36.9%	49.4%

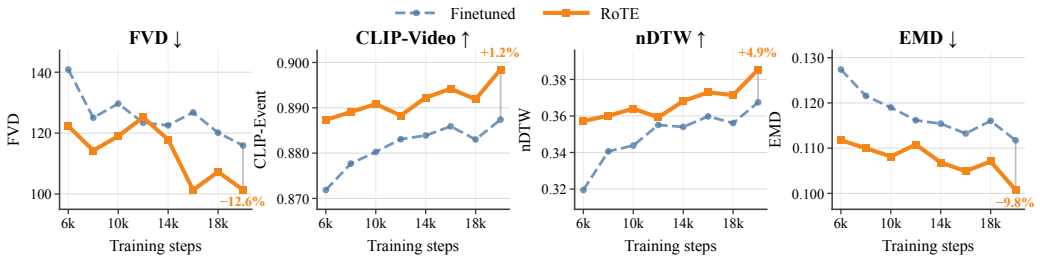

Figure 5: Performance trends during training with and without **TIE**. **TIE** leads to faster improvement across both visual and temporal metrics.

Table 3: Ablation study on *PexelsEvents*. The comparison between **NoRoPE**, **DoTE**, and **TIE** isolates the effect of interval-aware integration over prompt concatenation and boundary-only encoding.

Method	FID↓	FVD↓	CLIP-Event ↑	VQ↑	TC↑	DD↑	TA↑	FC↑
Base	59.68	357.51	0.226	2.73	2.78	2.58	2.68	2.65
NoRoPE	43.74	234.40	0.235	3.03	3.01	2.97	2.86	2.94
DoTE	43.84	234.79	0.241	3.05	3.01	3.01	2.88	2.94
RoTE	**42.53**	**217.29**	**0.246**	**3.10**	**3.05**	**3.06**	**2.92**	**2.99**
$\gamma = 2.0$	43.00	231.20	0.246	3.08	3.04	3.04	2.90	2.98
$\gamma = 8.0$	43.15	249.24	0.245	3.09	3.05	3.06	2.92	2.99

Case Study: Complex Multi-Subject Interaction. We evaluate **TIE** on gameplay data from *GameEvents* to test fine-grained action-centric event grounding in a high-dynamic multi-subject domain. Each sample contains structured event intervals for the player and boss. Unlike general web videos, this domain provides highly accurate event boundaries from game instrumentation, allowing us to test whether the model respects precise action timing. Figure 6 shows controlled prompt modifications applied to later event intervals while keeping earlier intervals fixed. The generated videos share the same early trajectory, then diverge according to the modified future event descriptions. In one sequence, the Tarnished rushes forward and attempts an attack but is interrupted by the boss. In another, the boss changes attack type, producing a different visual effect. In a third, the Tarnished rolls backward and avoids the hit. This demonstrates that **TIE** supports localized temporal editing: changing a future interval affects the intended future behavior while preserving the earlier trajectory in these examples, without a general guarantee of event–event consistency.

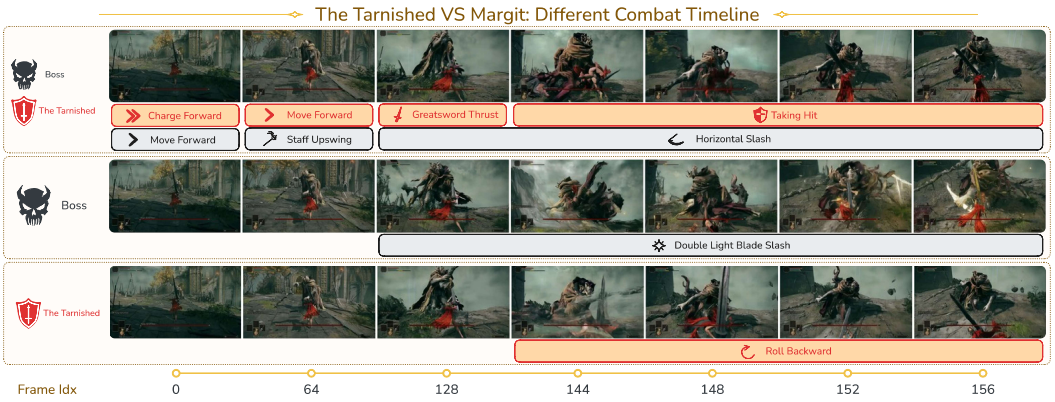

Figure 6: Generation Results on Game Data: In each row we show the frames and their actions. To illustrate the controllability enabled by **TIE**, we modify the target events and obtain different endings.

We also performed experiments on *RoboticsEvents*. Visual results can be found in Section D.3.

4.4 Ablation Study

Finally, we conduct minimal ablations to verify the components implied by the **TIE** formulation. Since **TIE** is a simple interval-encoding module rather than a multi-component system, we focus on the variants that directly correspond to the theory. **NoRoPE** removes temporal encoding and relies on prompt concatenation. **DoTE** encodes only interval boundaries using a Dirac-kernel variant, preserving start/end timestamps while discarding interior evidence. **RoTE** uses uniform-kernel integration over the full interval with normalization. We also test sensitivity to the scaling factor γ. The comparison between **NoRoPE** and **DoTE** shows that explicitly encoding temporal boundaries already improves temporal alignment. However, **DoTE** remains weaker than **TIE** because it ignores the interval interior. The full **RoTE** version achieves the best FVD and CLIP-Event, supporting interval integration in this ablation. The γ variants have similar semantic scores but different FVD; sensitivity depends on the metric.

4.5 Robustness to noisy temporal annotations.

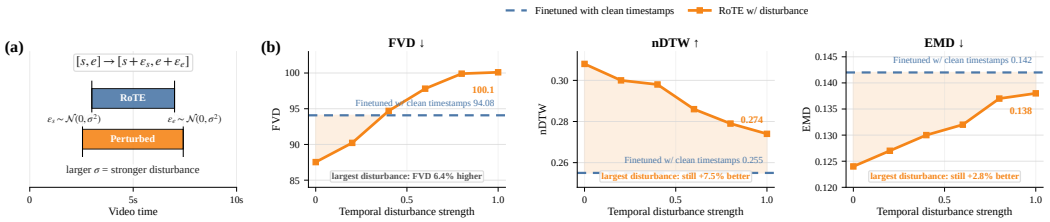

Figure 7: **Robustness to temporal annotation disturbance.** (a) We perturb the start and end timestamps of each **RoTE** interval with Gaussian noise: $[s, e] \to [s + \epsilon_s, e + \epsilon_e]$, where $\epsilon_s, \epsilon_e \sim \mathcal{N}(0, \sigma^2)$. (b) **RoTE** remains robust as the disturbance strength increases: even under the largest disturbance, it still achieves better temporal alignment than finetuning with clean timestamps, while only moderately increasing FVD.

Open-domain event annotations from VLMs, action detectors, or other perception systems can identify the correct semantics while misplacing start and end times by several frames or more. Reliable interval conditioning should therefore tolerate approximate boundaries rather than require frame-perfect supervision.

We perturb both endpoints of each RoTE interval with Gaussian noise and evaluate temporal alignment as the noise level increases (Figure 7). Even at the largest tested disturbance, RoTE outperforms the finetuning baseline trained with clean timestamps on nDTW and EMD. These results support using intervals as coarse temporal support with imperfect annotations, an important property for scaling event-conditioned training beyond manually curated data.

5 Conclusion

We propose Time Interval Encoding (TIE) to represent temporally extended events in video generation. Its uniform-kernel instance, RoTE, integrates rotary features over each interval while retaining the standard attention interface. The original experiments show improved human-verified and trajectory-level temporal grounding, faster convergence, and useful control under noisy boundary annotations on the primary backbone. Gameplay and robotics examples illustrate multi-subject interactions and future-event editing. These results motivate further study of interval-conditioned generation for interactive simulation and structured temporal control.

IMPACT STATEMENT

This paper presents work whose goal is to advance the field of machine learning, specifically focusing on **spatiotemporal modeling in video generation**. The proposed method contributes to the broader scientific community by **improving multi-subject control and temporal consistency in multi-event video generation**, with potential applications in **controllable video creation and simulation**.

AI USE STATEMENT

We used generative AI tools to assist with language editing, manuscript revision, integration of our experimental responses, and LaTeX formatting. These tools were not used to generate experimental measurements. The research workflow also uses language/vision-language models for structured video annotation, semantic verification, and prompt extension, as described in Appendices B.2 and D.

ETHICS STATEMENT

This work studies temporal control in video generation, with potential applications in content creation and simulation. Generated videos can also be used to misrepresent real events; temporal controllability does not establish physical correctness. The dataset combines web videos, gameplay recordings, and robotics data, whose provenance and construction are described in the appendix. Dataset reuse and release require attention to source licenses and privacy.

REPRODUCIBILITY STATEMENT

Section 3 specifies the encoding and calibration. Appendix A.1 provides proofs and an implementation sketch; the dataset and human-evaluation sections document annotation construction and temporal metrics. Appendix F records the additional interval baselines, kernels, duration/OOD evaluations, backbone comparisons, and measured runtime/memory settings. We will release the model weights, code, and data.

REFERENCES

NVIDIA: Niket Agarwal, Arslan Ali, Maciej Bala, Yogesh Balaji, Erik Barker, Tiffany Cai, Prithvijit Chattopadhyay, Yongxin Chen, Yin Cui, Yifan Ding, Daniel Dworakowski, Jiaojiao Fan, Michele Fenzi, Francesco Ferroni, Sanja Fidler, Dieter Fox, Songwei Ge, Yunhao Ge, Jinwei Gu, Siddharth Gururani, Ethan He, Jiahui Huang, Jacob Huffman, Pooya Jannaty, Jingyi Jin, Seung Wook Kim, Gergely Klár, Grace Lam, Shiyi Lan, Laura Leal-Taixe, Anqi Li, Zhaoshuo Li, Chen-Hsuan Lin, Tsung-Yi Lin, Huan Ling, Ming-Yu Liu, Xian Liu, Alice Luo, Qianli Ma, Hanzi Mao, Kaichun Mo, Arsalan Mousavian, Seungjun Nah, Sriharsha Niverty, David Page, Despoina Paschalidou, Zeeshan Patel, Lindsey Pavao, Morteza Ramezanali, Fitsum Reda, Xiaowei Ren, Vasanth Rao Naik Sabavat, Ed Schmerling, Stella Shi, Bartosz Stefaniak, Shitao Tang, Lyne Tchapmi, Przemek Tredak, Wei-Cheng Tseng, Jibin Varghese, Hao Wang, Haoxiang Wang, Heng Wang, Ting-Chun Wang, Fangyin Wei, Xinyue Wei, Jay Zhangjie Wu, Jiashu Xu, Wei Yang, Lin Yen-Chen, Xiaohui Zeng, Yu Zeng, Jing Zhang, Qinsheng Zhang, Yuxuan Zhang, Qingqing Zhao, and Artur Zolkowski. Cosmos world foundation model platform for Physical AI, 2025.

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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023.

Jake Bruce, Michael D. Dennis, Ashley Edwards, Jack Parker-Holder, Yuge Shi, Edward Hughes, Matthew Lai, Aditi Mavalankar, Richie Steigerwald, Chris Apps, Yusuf Aytar, Sarah Maria Elisabeth Bechtle, Feryal Behbahani, Stephanie C. Y. Chan, Nicolas Heess, Lucy Gonzalez, Simon Osindero, Sherjil Ozair, Scott Reed, Jingwei Zhang, Konrad Zolna, Jeff Clune, Nando De Freitas, Satinder Singh, and Tim Rocktäschel. Genie: generative interactive environments. In *Proceedings of the International Conference on Machine Learning*, 2024.

Emanuele Bugliarello, 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. In *Advances in Neural Information Processing Systems*, 2023.

Minghong Cai, Xiaodong Cun, Xiaoyu Li, Wenze Liu, Zhaoyang Zhang, Yong Zhang, Ying Shan, and Xiangyu Yue. DiTCtrl: exploring attention control in multi-modal diffusion transformer for tuning-free multi-prompt longer video generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2025.

Weifeng Chen, Jie Wu, Pan Xie, Hefeng Wu, Jiashi Li, Xin Xia, Xuefeng Xiao, and Liang Lin. Control-A-Video: controllable text-to-video generation with diffusion models. *arXiv preprint arXiv:2305.13840*, 2023.

AgiBot World Colosseum contributors. AgiBot world colosseum. `https://github.com/OpenDriveLab/AgiBot-World`, 2024.

Yilun Du, Sherry Yang, Bo Dai, Hanjun Dai, Ofir Nachum, Joshua B. Tenenbaum, Dale Schuurmans, and Pieter Abbeel. Learning universal policies via text-guided video generation. In *Advances in Neural Information Processing Systems*, 2023.

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

Xuan He, Dongfu Jiang, Ge Zhang, Max Ku, Achint Soni, Sherman Siu, Haonan Chen, Abhranil Chandra, Ziyan Jiang, Aaran Arulraj, Kai Wang, Quy Duc Do, Yuansheng Ni, Bohan Lyu, Yaswanth Narsupalli, Rongqi Fan, Zhiheng Lyu, Yuchen Lin, and Wenhu Chen. VideoScore: building automatic metrics to simulate fine-grained human feedback for video generation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, 2024.

Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, Günter Klambauer, and Sepp Hochreiter. GANs trained by a two time-scale update rule converge to a local Nash equilibrium. In *Advances in Neural Information Processing Systems*, 2017. URL `https://api.semanticscholar.org/CorpusID:231697514`.

Anthony Hu, Lloyd Russell, Hudson Yeo, Zak Murez, George Fedoseev, Alex Kendall, Jamie Shotton, and Gianluca Corrado. GAIA-1: a generative world model for autonomous driving. *arXiv preprint arXiv:2309.17080*, 2023.

Siyuan Huang, Liliang Chen, Pengfei Zhou, Shengcong Chen, Zhengkai Jiang, Yue Hu, Yue Liao, Peng Gao, Hongsheng Li, Maoqing Yao, et al. EnerVerse: envisioning embodied future space for robotics manipulation. In *Advances in Neural Information Processing Systems*, 2025.

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

Zeyinzi Jiang, Zhen Han, Chaojie Mao, Jingfeng Zhang, Yulin Pan, and Yu Liu. VACE: all-in-one video creation and editing. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2025.

Diederik P. Kingma and Jimmy Ba. Adam: a method for stochastic optimization. In *International Conference on Learning Representations*, 2015.

Yixuan Li, Lei Chen, Runyu He, Zhenzhi Wang, Gangshan Wu, and Limin Wang. MultiSports: a multi-person video dataset of spatio-temporally localized sports actions. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021.

Han Lin, Abhay Zala, Jaemin Cho, and Mohit Bansal. VideoDirectorGPT: consistent multi-scene video generation via LLM-guided planning. In *Proceedings of the First Conference on Language Modeling*, 2024.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019. URL https://api.semanticscholar.org/CorpusID:3312944.

Haoyu Lu, Guoxing Yang, Nanyi Fei, Yuqi Huo, Zhiwu Lu, Ping Luo, and Mingyu Ding. VDT: general-purpose video diffusion transformers via mask modeling. In *International Conference on Learning Representations*, 2024.

Xin Ma, Yaohui Wang, Xinyuan Chen, Gengyun Jia, Ziwei Liu, Yuan-Fang Li, Cunjian Chen, and Yu Qiao. Latte: latent diffusion transformer for video generation. *Transactions on Machine Learning Research*, 2025.

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 *European Conference on Computer Vision*. Springer, 2024.

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

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 *International Conference on Learning Representations*, 2024.

Qwen Team. Qwen3 technical report, 2025.

Hiroaki Sakoe and Seibi Chiba. Dynamic programming algorithm optimization for spoken word recognition. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 1978.

Uriel Singer, Adam Polyak, Thomas Hayes, Xi Yin, Jie An, Songyang Zhang, Qiyuan Hu, Harry Yang, Oron Ashual, Oran Gafni, Devi Parikh, Sonal Gupta, and Yaniv Taigman. Make-A-Video: text-to-video generation without text-video data. In *International Conference on Learning Representations*, 2023.

Jianlin Su et al. Roformer: Enhanced transformer with rotary position embedding. *Neurocomputing*, 2024.

Thomas Unterthiner, Sjoerd van Steenkiste, Karol Kurach, Raphaël Marinier, Marcin Michalski, and Sylvain Gelly. Towards accurate generative models of video: a new metric and challenges. *ArXiv*, 2018. URL https://api.semanticscholar.org/CorpusID:54458806.

Dani Valevski, Yaniv Leviathan, Moab Arar, and Shlomi Fruchter. Diffusion models are real-time game engines. In *International Conference on Learning Representations*, 2025.

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 *International Conference on Learning Representations*, 2023.

Team Wan, Ang Wang, Baole Ai, Bin Wen, Chaojie Mao, Chen-Wei Xie, Di Chen, Feiwu Yu, Haiming Zhao, Jianxiao Yang, Jianyuan Zeng, Jiayu Wang, Jingfeng Zhang, Jingren Zhou, Jinkai Wang, Jixuan Chen, Kai Zhu, Kang Zhao, Keyu Yan, Lianghua Huang, Mengyang Feng, Ningyi Zhang, Pandeng Li, Pingyu Wu, Ruihang Chu, Ruili Feng, Shiwei Zhang, Siyang Sun, Tao Fang, Tianxing Wang, Tianyi Gui, Tingyu Weng, Tong Shen, Wei Lin, Wei Wang, Wei Wang, Wenmeng Zhou, Wente Wang, Wenting Shen, Wenyuan Yu, Xianzhong Shi, Xiaoming Huang, Xin Xu, Yan Kou, Yangyu Lv, Yifei Li, Yijing Liu, Yiming Wang, Yingya Zhang, Yitong Huang, Yong Li, You Wu, Yu Liu, Yulin Pan, Yun Zheng, Yuntao Hong, Yupeng Shi, Yutong Feng, Zeyinzi Jiang, Zhen Han, Zhi-Fan Wu, and Ziyu Liu. Wan: open and advanced large-scale video generative models. *arXiv preprint arXiv:2503.20314*, 2025.

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*, 2023a.

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. In *Advances in Neural Information Processing Systems*, 2023b.

Yiping Wang, Xuehai He, Kuan Wang, Luyao Ma, Jianwei Yang, Shuohang Wang, Simon Shaolei Du, and Yelong Shen. Is your world simulator a good story presenter? a consecutive events-based benchmark for future long video generation. *arXiv preprint arXiv:2412.16211*, 2024.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Ed H. Chi, F. Xia, Quoc Le, and Denny Zhou. Chain-of-thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems*, 2022. URL https://api.semanticscholar.org/CorpusID:246411621.

Ziyi Wu, Aliaksandr Siarohin, Willi Menapace, Ivan Skorokhodov, Yuwei Fang, Varnith Chordia, Igor Gilitschenski, and Sergey Tulyakov. Mind the time: temporally-controlled multi-event video generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2025.

Zhifei Xie, Daniel Tang, Dingwei Tan, Jacques Klein, Tegawend F Bissyand, and Saad Ezzini. DreamFactory: pioneering multi-scene long video generation with a multi-agent framework. *arXiv preprint arXiv:2408.11788*, 2024.

Sherry Yang, Yilun Du, Seyed Kamyar Seyed Ghasemipour, Jonathan Tompson, Leslie Pack Kaelbling, Dale Schuurmans, and Pieter Abbeel. Learning interactive real-world simulators. In *International Conference on Learning Representations*, 2024a.

Shiyuan Yang, Liang Hou, Haibin Huang, Chongyang Ma, Pengfei Wan, Di Zhang, Xiaodong Chen, and Jing Liao. Direct-a-Video: customized video generation with user-directed camera movement and object motion. In *ACM SIGGRAPH 2024 Conference Papers*, 2024b.

Shuai Yang, Wei Huang, Ruihang Chu, Yicheng Xiao, Yuyang Zhao, Xianbang Wang, Muyang Li, Enze Xie, Yingcong Chen, Yao Lu, et al. LongLive: real-time interactive long video generation. In *International Conference on Learning Representations*, 2026.

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

Shengming Yin, Chenfei Wu, Huan Yang, Jianfeng Wang, Xiaodong Wang, Minheng Ni, Zhengyuan Yang, Linjie Li, Shuguang Liu, Fan Yang, Jianlong Fu, Ming Gong, Lijuan Wang, Zicheng Liu, Houqiang Li, and Nan Duan. NUWA-XL: diffusion over diffusion for eXtremely long video generation. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics*, 2023.

Hongyu Zhang, Yufan Deng, Zilin Pan, Peng-Tao Jiang, Bo Li, Qibin Hou, Zhiyang Dou, Zhen Dong, and Daquan Zhou. TS-Attn: temporal-wise separable attention for multi-event video generation. In *International Conference on Learning Representations*, 2026.

Mingzhe Zheng, Yongqi Xu, Haojian Huang, Xuran Ma, Yexin Liu, Wenjie Shu, Yatian Pang, Feilong Tang, Qifeng Chen, Harry Yang, et al. VideoGen-of-Thought: step-by-step generating multi-shot video with minimal manual intervention. *arXiv preprint arXiv:2412.02259*, 2024.

Zhou Zhi-Min. Pexels-400k: a large-scale dataset for video and image generation. https://huggingface.co/datasets/jovianzm/Pexels-400k, 2023.

Siyuan Zhou, Yilun Du, Jiaben Chen, Yandong Li, Dit-Yan Yeung, and Chuang Gan. RoboDreamer: learning compositional world models for robot imagination. In *Proceedings of the International Conference on Machine Learning*, 2024.

Fangqi Zhu, Hongtao Wu, Song Guo, Yuxiao Liu, Chilam Cheang, and Tao Kong. IRASim: a fine-grained world model for robot manipulation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2025.

A Details and Proofs for **TIE**

A.1 Proofs for Section 3

This subsection provides the full formalization and proofs for the principle-based characterization stated in Section 3.1, and proves Theorems A.2 and A.3.

Definition A.1 (Formal principles for interval-aware scores)**.** Let $s_{\mathrm{rope}}(\mathbf{q}, \mathbf{k}; t_q, \tau)$ denote the standard point-to-point **RoPE** score in Equation (2). We say that an interval-aware score $s(\mathbf{q}, \mathbf{k}; t_q, I)$ over an interval $I = [t^s, t^e]$ satisfies the following two principles:

Temporal Integrability, i.e., There exists a probability μ_I supported on I with $\mu_I(I) = 1$ such that the *raw* interval score is obtained by integrating point-wise **RoPE** scores over the temporal support of the interval,

$$\bar{s}(\mathbf{q}, \mathbf{k}; t_q, I) := \mathbb{E}_{\tau \sim \mu_I}[s_{\mathrm{rope}}(\mathbf{q}, \mathbf{k}; t_q, \tau)]. \tag{A1}$$

Duration Invariance, i.e., The final interval-aware score uses a positive scalar calibration of the probability expectation; unit expected center self-overlap specifies this scalar for the uniform kernel,

$$s(\mathbf{q}, \mathbf{k}; t_q, I) := \frac{1}{C(\mu_I)} \bar{s}(\mathbf{q}, \mathbf{k}; t_q, I), \qquad C(\mu_I) > 0, \tag{A2}$$

where $C(\mu_I)$ depends only on the interval measure μ_I and is independent of $\mathbf{q}$, $\mathbf{k}$, and t_q.

When both conditions hold, we call s a *principle-consistent* point-to-interval score.

Theorem A.2 (Uniform-kernel **TIE** yields **RoTE**)**.** *For the uniform kernel $\mu_I = \mathcal{U}([t^s, t^e])$ with interval center $c = (t^s + t^e)/2$ and radius $r = (t^e - t^s)/2$, assuming the stated bilinear form, unit expected center self-overlap, and $C_r > 0$, the encoder takes the closed form*

$$\mathbf{R}_{c,r} = \mathrm{diag}(\mathbf{A}_{1,c,r}, \mathbf{A}_{2,c,r}, \ldots, \mathbf{A}_{d/2,c,r}), \tag{A3}$$

and therefore reduces to the normalized interval encoder

$$\mathrm{RoTE}(\boldsymbol{k}, c, r) = \frac{1}{C_r} \mathbf{R}_{c,r} \boldsymbol{k}. \tag{A4}$$

Consequently, for a query token $\boldsymbol{q}_i$ at time m_i and a key token $\boldsymbol{k}_j$ associated with interval $I_j = [t_j^s, t_j^e]$, the **TIE** *attention score becomes*

$$\boldsymbol{s}_{i,j} \propto \mathrm{RoPE}(\boldsymbol{q}_i, m_i)^\top \mathrm{RoTE}(\boldsymbol{k}_j, c_j, r_j), \qquad c_j = \frac{t_j^s + t_j^e}{2},\ r_j = \frac{t_j^e - t_j^s}{2}. \tag{A5}$$

Proof of Theorem A.2. Set $\mu_I = \mathcal{U}([t^s, t^e])$ and write the interval as $I = [c - r, c + r]$. By Equation (4),

$$\mathrm{TIE}(\mathbf{k}, I) = \frac{1}{C_r} \frac{1}{2r} \int_{c-r}^{c+r} \mathbf{RoPE}(\mathbf{k}, \tau) \mathrm{d}\tau. \tag{A6}$$

For each frequency θ,

$$\frac{1}{2r} \int_{c-r}^{c+r} e^{\mathrm{i}\theta\tau} \mathrm{d}\tau = e^{\mathrm{i}\theta c} \,\mathrm{sinc}(\theta r), \tag{A7}$$

so every 2×2 **RoPE** block averages to $\mathbf{A}_{i,c,r} = \mathrm{sinc}(\theta_i r) \mathbf{A}_{i,c}$. Stacking the blocks yields the final results.

To fix the self-overlap calibration, we additionally require unit expected self-overlap at the interval center:

$$1 = \mathbb{E}_{\mathbf{q} \sim \mathcal{U}(S^{d-1})} \left[\mathrm{RoPE}(\mathbf{q}, c)^\top \mathrm{RoTE}(\mathbf{q}, c, r) \right]. \tag{A8}$$

The reference distribution does not assert isotropy of learned queries. Using the isotropy identity $\mathbb{E}_{\mathbf{q} \sim \mathcal{U}(S^{d-1})}[\mathbf{q}^\top M \mathbf{q}] = \frac{1}{d} \mathrm{tr}(M)$, we obtain

$$1 = \frac{1}{C_r} \frac{2}{d} \sum_{\ell=1}^{d/2} \mathrm{sinc}(\theta_\ell r), \tag{A9}$$

Substituting the resulting encoder into the **TIE** attention score yields the attention score Equation (5) in Equation (7) □

Theorem A.3 (Dirac-kernel **TIE** yields **DoTE**). *If the two channel groups are calibrated at their respective endpoints and instantiated with Dirac kernels concentrated at the interval boundaries, i.e., $\mu_{I,s} = \delta_{t^s}$ and $\mu_{I,e} = \delta_{t^e}$ on the two channel groups of $\boldsymbol{k} = \boldsymbol{k}^{(s)} \oplus \boldsymbol{k}^{(e)}$, then the resulting interval encoder is exactly the boundary-only form* **DoTE**:

$$\mathrm{DoTE}(\boldsymbol{k}, I) = \mathbf{Concat}(\mathrm{RoPE}(\boldsymbol{k}^{(s)}, t^s), \mathrm{RoPE}(\boldsymbol{k}^{(e)}, t^e)). \tag{A10}$$

Proof of Theorem A.3. Let $\mathbf{k} = \mathbf{k}^{(s)} \oplus \mathbf{k}^{(e)}$ and instantiate **TIE** on the two channel groups with the Dirac kernels $\mu_{I,s} = \delta_{t^s}$ and $\mu_{I,e} = \delta_{t^e}$. Calibrating each channel group at its own endpoint gives unit self-overlap and hence $C(\mu_{I,s}) = C(\mu_{I,e}) = 1$. Applying **TIE** to the two subspaces gives

$$\mathrm{TIE}_s(\mathbf{k}^{(s)}, I) = \mathrm{RoPE}(\mathbf{k}^{(s)}, t^s), \qquad \mathrm{TIE}_e(\mathbf{k}^{(e)}, I) = \mathrm{RoPE}(\mathbf{k}^{(e)}, t^e). \tag{A11}$$

Concatenating the two channel groups yields

$$\mathrm{TIE}(\mathbf{k}, I) = \mathbf{Concat}(\mathrm{RoPE}(\mathbf{k}^{(s)}, t^s), \mathrm{RoPE}(\mathbf{k}^{(e)}, t^e)) = \mathrm{DoTE}(\mathbf{k}, I), \tag{A12}$$

which is exactly the DoTE. Since the construction keeps only two boundary atoms, it is a boundary-only specialization of **TIE** and does not average over the interval interior. □

A central requirement for large-scale event-conditioned video pretraining is tolerance to imperfect temporal boundaries. In small curated benchmarks, event intervals may be manually verified with high precision. At scale, however, event-time annotations are more likely to come from VLMs, action detectors, captioning models, or other automatic perception systems. These annotators can provide useful event semantics, but their predicted start and end times are inevitably noisy. Therefore, an interval-aware temporal encoding should not depend on frame-perfect endpoints. It should be able to treat noisy intervals as coarse temporal support for event grounding. The following theorem formalizes this property for **RoTE**. The key distinction is that **RoPE** and **DoTE** encode events through pointwise temporal phases: **RoPE** collapses an event to a single timestamp, while **DoTE** places all temporal information on the two boundaries. Consequently, perturbing the timestamp or either endpoint directly perturbs the encoded phase. In contrast, **RoTE** represents an event by a duration-normalized integral of **RoPE** features over the interval. Boundary perturbations therefore affect only the marginal portion of the integration domain, yielding a robustness behavior controlled by the relative boundary disturbance.

Theorem A.4 (Robustness of **RoTE** to bounded timestamp noise). *Define the unscaled interval-attention score*

$$\boldsymbol{s}_{i,j}(c, r) := \mathrm{RoPE}(\boldsymbol{q}_i, m_i)^\top \mathrm{RoTE}(\boldsymbol{k}_j, c, r), \qquad \mathrm{RoTE}(\boldsymbol{k}, c, r) = C_r^{-1} \mathbf{R}_{c,r} \boldsymbol{k}. \tag{A13}$$

For event $I_j = [t_j^s, t_j^e]$, let $c_j = (t_j^s + t_j^e)/2$ and $r_j = (t_j^e - t_j^s)/2 > 0$. Suppose its observed endpoints are

$$\widetilde{t}_j^s = t_j^s + \epsilon_j^s, \qquad \widetilde{t}_j^e = t_j^e + \epsilon_j^e, \qquad |\epsilon_j^s|, |\epsilon_j^e| \le \delta < r_j \quad \textit{a.s.} \tag{A14}$$

Write

$$\Delta c_j := \frac{\epsilon_j^s + \epsilon_j^e}{2}, \qquad \Delta r_j := \frac{\epsilon_j^e - \epsilon_j^s}{2}, \qquad \widetilde{c}_j = c_j + \Delta c_j, \qquad \widetilde{r}_j = r_j + \Delta r_j. \tag{A15}$$

Assume the normalization does not degenerate on the perturbed radius range,

$$C_{\min,j} := \inf_{\rho \in [r_j - \delta, r_j + \delta]} |C_\rho| > 0. \tag{A16}$$

Then

$$|\boldsymbol{s}_{i,j}(\widetilde{c}_j, \widetilde{r}_j) - \boldsymbol{s}_{i,j}(c_j, r_j)| \le \frac{\|\boldsymbol{q}_i\| \, \|\boldsymbol{k}_j\| \, \delta}{r_j - \delta} \left(\frac{3}{C_{\min,j}} + \frac{2}{C_{\min,j}^2} \right). \tag{A17}$$

Therefore, if $\delta \le r_j/2$ and $C_{\min,j}$ is bounded away from zero, **RoTE** *has almost-sure and hence expected noise sensitivity $O(\|\boldsymbol{q}_i\| \, \|\boldsymbol{k}_j\| \, \delta / r_j)$. By contrast, point-wise* **RoPE** *and boundary-only* **DoTE** *have local worst-case timestamp sensitivity $O(\|\boldsymbol{q}_i\| \, \|\boldsymbol{k}_j\| \, \theta_{\max} \delta)$, where $\theta_{\max} := \max_\ell \theta_\ell$, with no decay in the interval radius r_j.*

Proof of Theorem A.4. Let

$$\Phi(c,r) := C_r^{-1}\mathbf{R}_{c,r}, \qquad \mathbf{s}_{i,j}(c,r) = \mathrm{RoPE}(\mathbf{q}_i, m_i)^\top \Phi(c,r)\mathbf{k}_j. \tag{A18}$$

Since **RoPE** is an orthogonal block rotation, $\|\mathrm{RoPE}(\mathbf{q}_i, m_i)\| = \|\mathbf{q}_i\|$.

We first bound the derivatives of Φ. For each frequency block,

$$\mathbf{R}_{c,r}^{(\ell)} = \mathrm{sinc}(\theta_\ell r)\begin{bmatrix}\cos(\theta_\ell c) & -\sin(\theta_\ell c)\\ \sin(\theta_\ell c) & \cos(\theta_\ell c)\end{bmatrix}. \tag{A19}$$

The rotation matrix has operator norm one and $|\,\mathrm{sinc}(x)| \le 1$, so

$$\|\mathbf{R}_{c,r}\|_{\mathrm{op}} \le 1. \tag{A20}$$

For $r > 0$, blockwise differentiation gives

$$\|\partial_c \mathbf{R}_{c,r}\|_{\mathrm{op}} = \max_\ell \theta_\ell |\,\mathrm{sinc}(\theta_\ell r)| = \max_\ell \frac{|\sin(\theta_\ell r)|}{r} \le \frac{1}{r}. \tag{A21}$$

Similarly, since

$$\theta\ \mathrm{sinc}'(\theta r) = \frac{\cos(\theta r) - \mathrm{sinc}(\theta r)}{r}, \tag{A22}$$

we have

$$\|\partial_r \mathbf{R}_{c,r}\|_{\mathrm{op}} = \max_\ell |\theta_\ell\,\mathrm{sinc}'(\theta_\ell r)| \le \frac{2}{r}. \tag{A23}$$

Moreover,

$$C_r = \frac{2}{d}\sum_{\ell=1}^{d/2}\mathrm{sinc}(\theta_\ell r) \quad\Longrightarrow\quad |C_r'| \le \frac{2}{d}\sum_{\ell=1}^{d/2}|\theta_\ell\,\mathrm{sinc}'(\theta_\ell r)| \le \frac{2}{r}. \tag{A24}$$

On $\rho \in [r_j - \delta, r_j + \delta]$, we have $|C_\rho| \ge C_{\min,j}$ and $\rho \ge r_j - \delta$. Hence

$$\|\partial_c \Phi(c,\rho)\|_{\mathrm{op}} \le \frac{1}{C_{\min,j}(r_j - \delta)}, \tag{A25}$$

and, using $\partial_r \Phi = C_r^{-1}\partial_r \mathbf{R}_{c,r} - C_r' C_r^{-2}\mathbf{R}_{c,r}$,

$$\|\partial_r \Phi(c,\rho)\|_{\mathrm{op}} \le \frac{1}{r_j - \delta}\left(\frac{2}{C_{\min,j}} + \frac{2}{C_{\min,j}^2}\right). \tag{A26}$$

Now fix one realization of the endpoint perturbation. Write $x = (c_j, r_j)$ and $\Delta = (\Delta c_j, \Delta r_j)$. Since $|\Delta r_j| \le \delta$, every point on the line segment $x + \tau\Delta$, $\tau \in [0,1]$, has radius in $[r_j - \delta, r_j + \delta]$. By the fundamental theorem of calculus in the finite-dimensional matrix space,

$$\begin{aligned}\Phi(x+\Delta) - \Phi(x) &= \int_0^1 D\Phi(x+\tau\Delta)[\Delta]\,d\tau \\ &= \int_0^1 \left(\Delta c_j\,\partial_c\Phi(x+\tau\Delta) + \Delta r_j\,\partial_r\Phi(x+\tau\Delta)\right)\,d\tau.\end{aligned} \tag{A27}$$

Taking operator norms and using the triangle inequality gives

$$\begin{aligned}\|\Phi(x+\Delta) - \Phi(x)\|_{\mathrm{op}} &\le |\Delta c_j| \sup_{\tau\in[0,1]} \|\partial_c\Phi(x+\tau\Delta)\|_{\mathrm{op}} \\ &\quad + |\Delta r_j| \sup_{\tau\in[0,1]} \|\partial_r\Phi(x+\tau\Delta)\|_{\mathrm{op}} \\ &\le \frac{1}{r_j - \delta}\left[\frac{|\Delta c_j|}{C_{\min,j}} + \left(\frac{2}{C_{\min,j}} + \frac{2}{C_{\min,j}^2}\right)|\Delta r_j|\right].\end{aligned} \tag{A28}$$

This is the operator-valued mean-value inequality; it does not require an equality-form mean value theorem for the operator norm.

Therefore,

$$\begin{aligned}
&|\mathbf{s}_{i,j}(\widetilde{c}_j, \widetilde{r}_j) - \mathbf{s}_{i,j}(c_j, r_j)| \\
&\quad = \left|\mathrm{RoPE}(\mathbf{q}_i, m_i)^\top \left(\Phi(\widetilde{c}_j, \widetilde{r}_j) - \Phi(c_j, r_j)\right) \mathbf{k}_j\right| \\
&\quad \le \|\mathbf{q}_i\| \, \|\mathbf{k}_j\| \, \|\Phi(\widetilde{c}_j, \widetilde{r}_j) - \Phi(c_j, r_j)\|_{\mathrm{op}} \,. \qquad \text{(A29)}
\end{aligned}$$

Since $|\epsilon_j^s|, |\epsilon_j^e| \le \delta$ almost surely, we also have $|\Delta c_j|, |\Delta r_j| \le \delta$, which gives Equation (A17).

Finally, for point-wise **RoPE**, let $\mathbf{R}_t$ be the standard RoPE rotation. Blockwise differentiation gives

$$\left\| \frac{d}{dt} \mathbf{R}_t \right\|_{\mathrm{op}} = \max_\ell \theta_\ell = \theta_{\max}. \qquad \text{(A30)}$$

Hence the same line-integral argument yields

$$\|\mathbf{R}_{t+\epsilon} - \mathbf{R}_t\|_{\mathrm{op}} \le \theta_{\max} |\epsilon|, \qquad \text{(A31)}$$

and this local worst-case Lipschitz scale is independent of any interval radius. The boundary-only **DoTE** encoder is a concatenation of two point-wise encoders at t^s and t^e, so it has the same order of timestamp sensitivity and no $1/r_j$ decay. □

Remark A.5 (Interpretation of the robustness bound). The theorem shows that the timestamp-noise sensitivity of **RoTE** is controlled by the relative boundary error δ/r_j when the normalization is non-degenerate. This relative scaling is the key difference from point-wise or boundary-only encodings. Since **RoTE** aggregates positional evidence over the full temporal support of the event, perturbing the start or end time only changes the marginal portion of the interval representation. As a result, the effect of boundary noise is averaged over the event duration, and the bound decreases with duration provided normalization remains uniformly bounded away from zero. In contrast, point-wise **RoPE** collapses the event to a single timestamp, while boundary-only **DoTE** places all temporal information on the two endpoints; in both cases, timestamp noise directly perturbs the encoded phase and does not decay with the interval radius.

Remark A.6 (Low-pass smoothing from the closed-form interval integral). The robustness of $\mathbf{RoTE}$ can also be understood from the closed-form frequency response of the interval integral. For each RoPE frequency θ_ℓ, the uniform-kernel interval encoder multiplies the corresponding rotation block by $\mathrm{sinc}(\theta_\ell r)$. Therefore, perturbations of the interval center are attenuated at high temporal frequencies:

$$\theta_\ell \, |\mathrm{sinc}(\theta_\ell r)| = \left| \frac{\sin(\theta_\ell r)}{r} \right| \le \frac{1}{r}. \qquad \text{(A32)}$$

Thus, up to the non-degenerate normalization factor, the local sensitivity of $\mathbf{RoTE}$ to center perturbations decays with the interval radius. For small intervals, this behavior reduces to the point-wise **RoPE** regime; for long intervals, the interval integral acts as a temporal low-pass filter and suppresses high-frequency phase noise. This smoothing effect is absent in point-wise $\mathbf{RoPE}$ and boundary-only $\mathbf{DoTE}$, whose temporal information is concentrated on one timestamp or two endpoints and therefore does not become less sensitive as the interval becomes longer.

Remark A.7 (Why learned interval embeddings do not provide the same guarantee). A learned interval embedding could in principle become robust to noisy boundaries if it is trained with sufficiently diverse boundary perturbations. However, this robustness would be distribution-dependent: it need not follow this bound without assumptions on the architecture or its derivatives. Other learned embeddings may also admit robustness guarantees. In contrast, the robustness of $\mathbf{RoTE}$ is architectural. The sinc attenuation is built into the closed-form interval representation itself, so the smoothing effect does not rely on learning a particular noise distribution or adding explicit boundary-noise augmentation. This property is especially important for large-scale pretraining, where event intervals may be generated by different VLMs or detectors whose boundary errors vary across domains.

Remark A.8 (Implication for large-scale event-conditioned pretraining). This robustness is important for scaling interval-aware video generation beyond small curated datasets. In real-world data collection, event intervals are often obtained from VLMs, action detectors, captioning models, or other neural annotators. Such pipelines can identify the correct event semantics at scale, but their start and end timestamps are inevitably approximate. Therefore, robustness to boundary noise is not merely a desirable property of **RoTE**; it is a core requirement for using interval-conditioned objectives in large-scale pretraining. The δ/r_j sensitivity indicates that **RoTE** can use noisy intervals as coarse temporal support, rather than depending on frame-perfect endpoint supervision.

A.2 ALGORITHM

Here is the demonstrative code for **TIE**:

```python
def RoTE(start, end, theta=10000.0, scaling_factor=4.0, alpha=1.0, dim=128):
  # Assuming start, end are torch.Tensor of shape (b, t)
  center = (start + end) * scaling_factor / 2.0
  radius = (end - start) * scaling_factor / 2.0
  freqs = 1.0 / (theta ** (torch.arange(0, dim, 2).double() / dim)).to(start.device)
  theta = torch.einsum("bi,j->bij", center, freqs)
  cos, sin = theta.cos(), theta.sin()
  phi = torch.einsum("bi,j->bij", radius, freqs)
  sincs = torch.sinc(alpha * phi/torch.pi)
  sincs = sincs / torch.mean(sincs, dim=-1, keepdim=True)
  cos, sin = cos * sincs, sin * sincs
  return cos[:, :, None, :], sin[:, :, None, :]

def RoTE_apply(x, cos, sin, num_heads):
  x = rearrange(x, "b s (n u h) -> b s n u h", n=num_heads, u=2)
  x_out = torch.cat([x[:, :, :, 0, :] * cos - x[:, :, :, 1, :] * sin, x[:, :, :, 0, :] * sin + x[:, :, :, 1, :] * cos], dim=-1).flatten(2)
  return x_out.to(x.dtype)
```

A.3 **RoTE** VISUALIZATION

Figure A1 & A2 show a visualization case for **RoTE** decaying. Events with longer duration decay slower along the time interval.

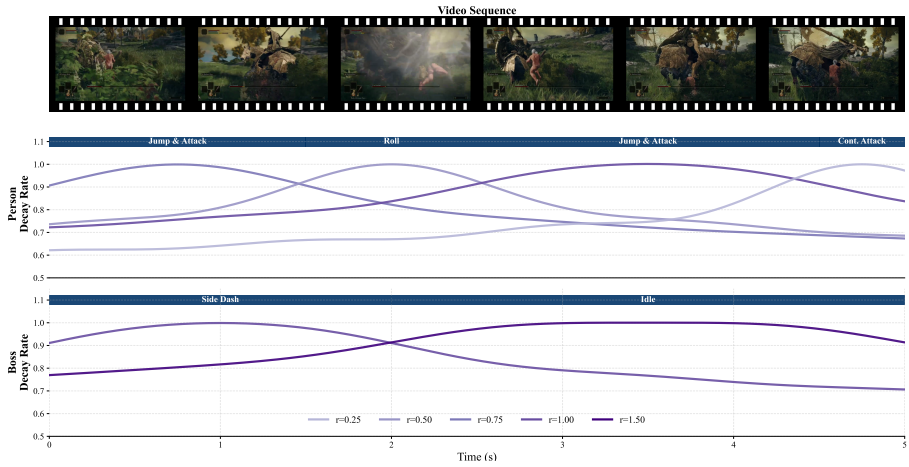

Figure A1: **RoTE** Visualization: The curves show the decaying effect of each event.

B DATASET

B.1 DATA CONSTRUCTION VIA SELF-REFLECTIVE AGENT

Since interval-based modeling requires temporally precise and structured event supervision, we construct a large-scale dataset using a self-reflective annotation agent.

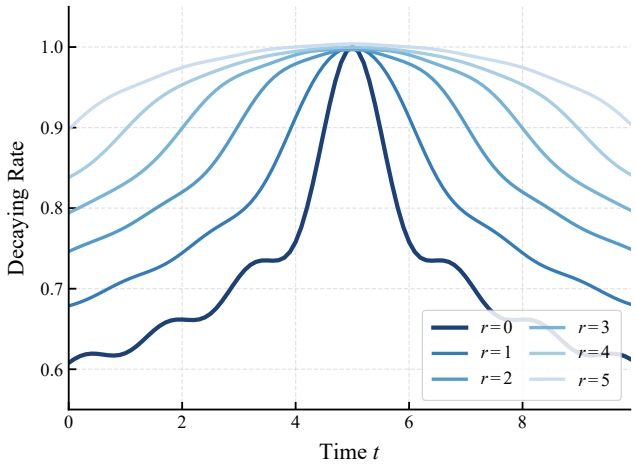

Figure A2: **RoTE** Decaying Rate: We visualize the relative decaying rate of our **RoTE** when $\gamma = 4.0$. For a video clip of 10s long, we fix the middle point of the event at 5.0s and vary the radius r of event. When $r = 0$, **RoTE** generally reduces to normal **RoPE**. While a larger r dampens the decaying rate. For a long event that sustains for 10 seconds ($r = 5$), the decaying rate remains above 0.9 during the whole event duration.

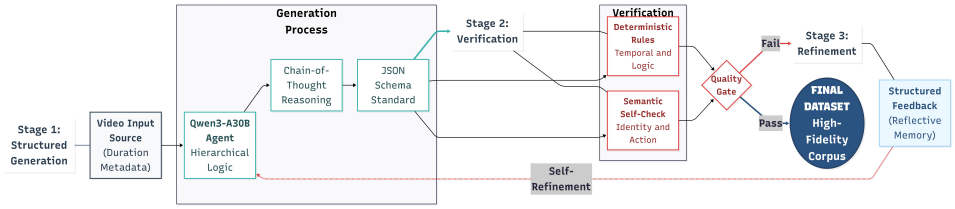

Figure A3: **Overview of the Self-Reflective Video Annotation Pipeline.** The system constructs high-fidelity datasets through three iterative stages: (1) **Structured Generation**, employing Qwen3-A30B with **CoT reasoning** and hierarchical decomposition under a strict **JSON schema**; (2) **Dual-Track Verification**, which enforces **deterministic constraints** (temporal/logical filters) and **semantic self-checks** to ensure identity and action fidelity; and (3) a **Refinement Loop**, where detected failures trigger **agentic self-correction** via structured error feedback. This workflow effectively eliminates temporal hallucinations and ensures the plausibility of the final corpus.

B.1.1 The Structured Generation

We employ a large-scale vision-language model (Qwen3-A30B (Qwen Team, 2025)) to generate dense video annotations. To ensure data parseability and consistency, we enforce a strict JSON schema rather than allowing free-form textual output. The prompt is designed to guide the model through a hierarchical decomposition of the video content: it must first explicitly define participant identities (e.g., tracking visual attributes like clothing) and global scene context before generating detailed event timelines. Additionally, we leverage Chain-of-Thought (CoT) prompting (Wei et al., 2022), requiring the model to explicitly reason about visual anchors and temporal continuity before finalizing the event logs. Details about the prompt can be found in Section B.2.

B.1.2 The Verification Criteria

All verification criteria are designed to enforce consistency with the interval-based event formulation introduced in Section 3.1. To mitigate the common issue of temporal **hallucination** in VLMs, we implement a verification mechanism anchored by ground-truth metadata. Crucially, we explicitly inject the precise video duration (extracted via FFmpeg) into the system prompt. This serves as a hard boundary, preventing the model from generating events that exceed the physical limits of the video.

Building on this ground truth, we enforce a set of deterministic constraints:

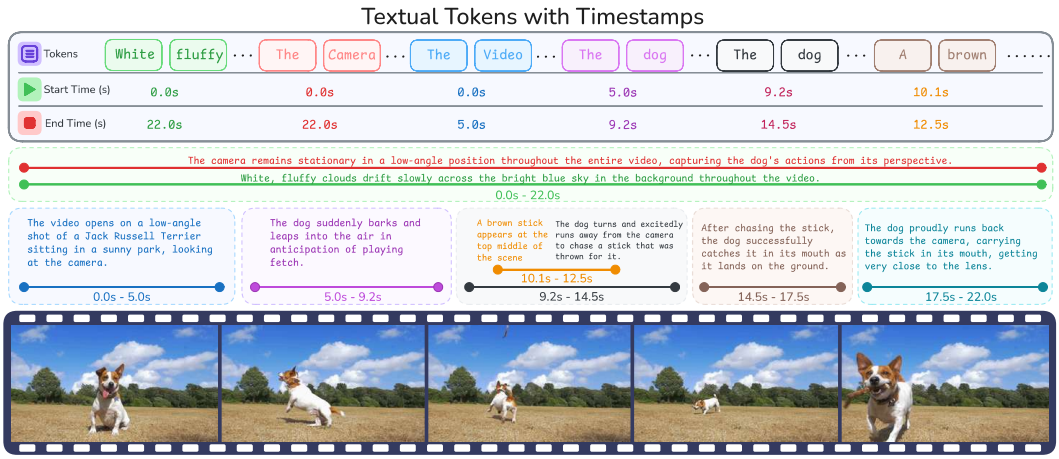

Figure A4: Textual Tokens with Timestamps: The captions for videos are structured and timestamped, which could be utilized to annotate every token with an accurate start time and end time. In this case, the caption describes the scene, the characters and objects, the background, behaviors and camera motions in detail. Every event is annotated with timestamps of start and end at the accuracy of 0.25s.

Temporal Alignment All timestamps are strictly quantized to a 0.25s grid. This discretization is specifically designed to align with the temporal granularity of our downstream video generation backbone (operating at 16 fps), ensuring that text-conditioned events map precisely to video frame blocks. Figure A4 gives an example of the timestamps.

Validity Checks We apply a series of logical filters to ensure the physical and semantic plausibility of the annotations. First, we enforce that every event's end time must strictly follow its start time, eliminating invalid negative durations. Second, to ensure clear action boundaries, we prohibit overlapping events within the same action track to ensure atomic action definition. This guarantees that distinct actions do not conflict in the timeline. Third, we impose a minimum duration threshold of 1.0s. This filter removes fleeting micro-actions that are too brief for stable video generation, ensuring that every annotated event represents a significant and renderable motion phase.

Semantic Consistency While rule-based checks ensure structural validity, they cannot detect semantic errors (e.g., hallucinated objects or identity swaps). To address this, we introduce a semantic self-verification step. In this phase, we feed the generated timeline back into the VLM alongside the original video, instructing the model to act as an independent critic. The VLM is tasked with verifying two critical aspects:

- **Identity Consistency** It checks whether the visual attributes described in the text (e.g., "person in the red shirt") remain consistent with the pixel-level visual information throughout the video, flagging any identity switches.
- **Action Fidelity** It assesses whether the captioned actions (e.g., "picking up a cup") accurately reflect the actual motion dynamics observed in the video frames. If the VLM detects a mismatch, it outputs a specific error description, which triggers the refinement loop.

This semantic verification step improves annotation reliability but is not required at inference time.

B.1.3 The Refinement Loop

This mechanism serves as the core of our agentic workflow, enabling the system to self-correct rather than simply discarding imperfect generations. When an annotation fails either the programmatic validity checks or the semantic self-verification, the specific error message (e.g., "Event 3 duration 0.5s is less than 1.0s threshold" or "Identity mismatch at 4.5s") is formulated as structured feedback. We inject this feedback directly into the prompt for the next inference cycle, explicitly instructing the VLM to reflect on its previous error.

B.2 CONSTRUCTION OF THE PEXELSEVENTS

Here is our annotation structure for PexelsEvents:

```
{"scene_clarity": "Clear|Cluttered|Chaotic",
"global_caption": {
"short_caption": "Concise summary of scene (NO PEOPLE/ACTION), ~20 words.",
"long_caption": "Detailed description of environment and static objects (NO PEOPLE), ~60 words.",
"visible_objects": ["list", "of", "all", "distinct", "objects", "visible"],
"start_time": 0.0,
"end_time": <duration:.2f>},
"participants": {
"participant_id": {
"short_description": "Short description starting with ID.",
"long_description": "Detailed description starting with ID, distinguishing features.",
"start_time": "float (Exact time of first appearance)",
"end_time": "float (Exact time of last disappearance)",
"timeline": [
{
"distinguishing_feature": "Visual check (e.g., 'red shirt').",
"start_time": float,
"end_time": float,
"is_interaction": boolean,
"interaction_with": ["other_id"],
"short_caption": "Verb + object...",
"long_caption": "Detailed phase log starting with ID. Max 50words. Include start/end positions and specific body part used."
}
]
}
},
"statistics": {
"video_quality_score": 1-5,
"max_events_count": int,
"participants_with_max_events": ["id"],
"is_suitable_for_learning": boolean
}
}
```

Figure A5 shows an example from PexelsEvents.

B.3 CONSTRUCTION OF THE GAMEEVENTS

We collect 128 hours of Elden Ring gameplay videos. Videos are recorded by OBS. All behaviors of player and bosses are recorded by reading the real-time animation ID from the game memory. Then we transform these records into 80k 10-second video clips with structured annotations. Figure A6 shows an example data containing high-dynamic fast-speed interactions of the player and the boss.

B.4 CONSTRUCTION OF THE ROBOTICSEVENTS

We select 86k 10-second video clips from Agibot-World dataset. Which covers several manipulation tasks (including some scenes with human intervention). We filter the recorded trajectories and split them into sub-pieces. We feed VLM with information of these trajectory segments to generate more accurate structured prompts that describe behaviors of each arm and gripper. Figure A7 shows an example data containing complex interactions of robot and human operator.

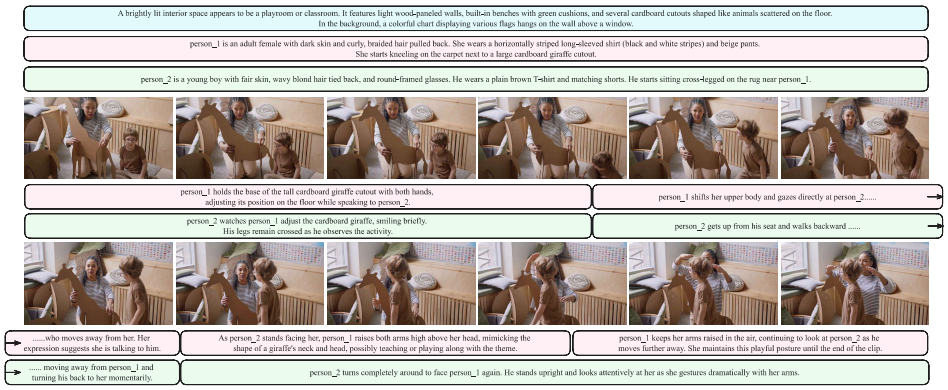

Figure A5: An example from *PexelsEvents*: global descriptions and event-level descriptions are provided.

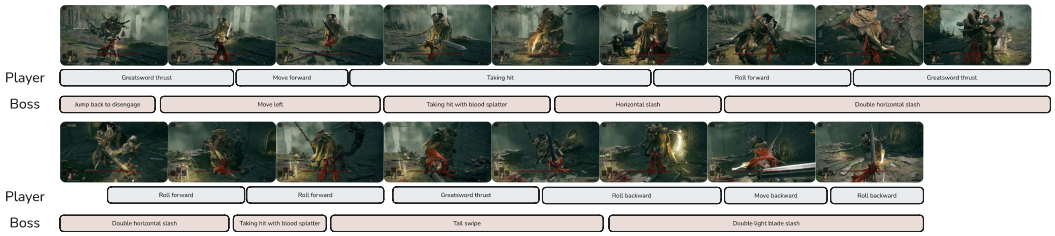

Figure A6: An example from *GameEvents*: representative animation sequences with detailed event annotations.

B.5 Dataset Statistics

We report several statistics about our *OmniEvents* dataset in Table A1. The Overlap Prob. is the fraction of events overlapping at least one other event, rather than the fraction of clips containing overlap. These three parts of *OmniEvents* covers from low-speed (*PexelsEvents*) to high-speed (*GameEvents*).

Table A1: Statistics of *OmniEvents*. **EC** and **ED** denote the average number of events per sample and the average duration per event. **TE**, **TD**, and **TL** denote the total number of events, the total duration of all events, and the total character length of all text prompts, respectively. **Overlap Prob.** denotes the probability of event overlap.

Dataset	Video Clips	EC	ED	TE	TD	TL	Overlap Prob.
PexelsEvents	253,903	4.72	3.67	1,197,973	4,391,589	164,639,876	68.0%
GameEvents	79,959	16.01	1.24	1,280,208	1,584,762	42,368,009	99.63%
RoboticsEvents	85,956	14.47	2.79	1,244,058	3,472,766	86,717,027	99.99%

C Human Evaluation

Human annotation protocol. All temporal metrics are computed from human-verified event annotations rather than automatic detectors. We randomly sample 100 prompts from the evaluation dataset. For each prompt, we generate one video using the finetuned baseline and one video using our method with TIE, resulting in 200 generated videos in total. Each prompt contains a set of requested events with specified target intervals. For every generated video, human annotators are shown the video and the corresponding prompt event list. For each requested event, they are asked to answer three questions: (1) whether the event occurs in the video; (2) how much the observed event start time deviates from the requested start time; and (3) how much the observed event end time deviates

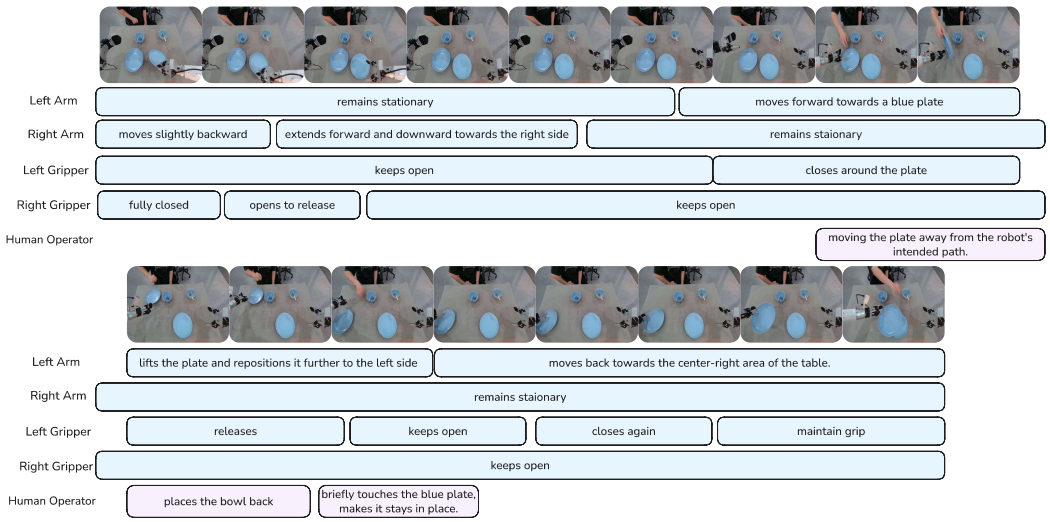

Figure A7: An example from *RoboticsEvents*: representative manipulation sequences with detailed event annotations.

from the requested end time. We employ 10 human annotators and compute the final metrics from the average of their responses. Figure A8 gives the annotation user interface.

Event representation. Each prompt contains a set of requested events $\mathcal{E} = \{e_i\}_{i=1}^{N}$. Each event e_i is specified by an event description, an actor, and a target time interval $[s_i, t_i]$. For each generated video, annotators provide a binary occurrence label $o_i \in \{0, 1\}$ indicating whether e_i appears in the video. If $o_i = 1$, annotators additionally provide the start-time bias b_i^s and end-time bias b_i^t with respect to the requested interval:

$$b_i^s = \hat{s}_i - s_i, \quad b_i^t = \hat{t}_i - t_i,$$

where $[\hat{s}_i, \hat{t}_i]$ is the observed event interval in the generated video.

Event Occurrence. Event Occurrence measures the probability that a requested event is realized in the generated video:

$$\mathrm{Occ} = \frac{1}{N} \sum_{i=1}^{N} o_i.$$

This metric captures the most basic requirement of temporal control: the model must first generate the requested event before its timing can be evaluated. We compute it over all requested prompt events, so missing events directly reduce the score.

Temporal Error. Temporal Error measures the temporal deviation between the requested interval and the human-observed interval. If an event occurs, we use the average absolute boundary deviation:

$$\mathrm{TE}_i = \frac{|b_i^s| + |b_i^t|}{2}.$$

If an event is missing, its boundary deviation is undefined. We therefore assign the requested event duration as a missing-event penalty:

$$\mathrm{TE}_i = t_i - s_i, \quad \text{if } o_i = 0.$$

The final Temporal Error is averaged over all requested events:

$$\mathrm{TE} = \frac{1}{N} \sum_{i=1}^{N} \mathrm{TE}_i.$$

This prevents a model from receiving an artificially low timing error by simply omitting difficult events. The event duration provides a natural penalty scale: missing a short event is less severe than missing a long event, while still being counted as a temporal failure.

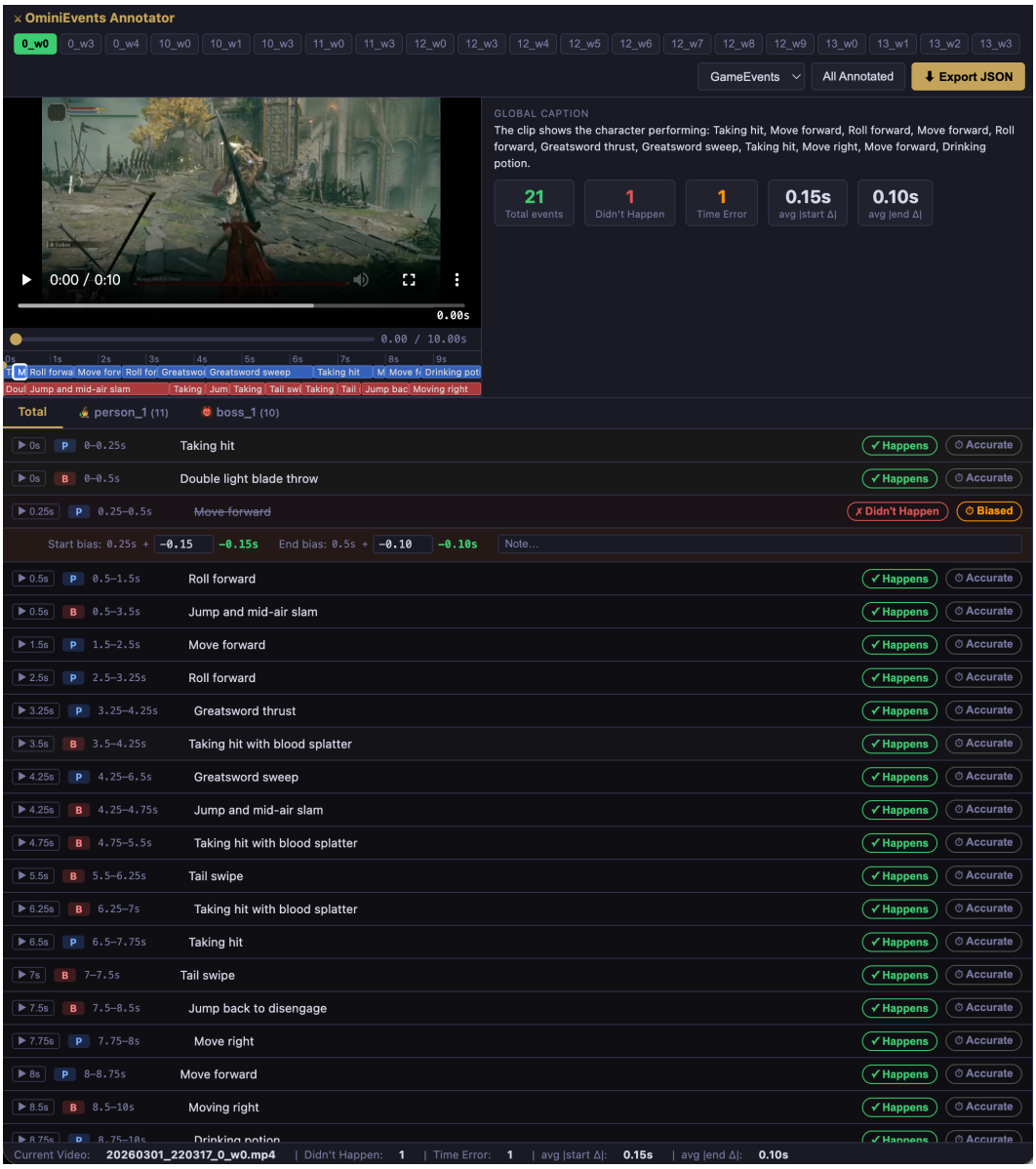

Figure A8: An example of the user interface for annotating temporal alignment.

Order Accuracy. Order Accuracy measures whether the generated video preserves the before/after relations specified by the prompt. For each pair of requested events (e_i, e_j) in the same prompt, we compare their target center times:

$$c_i = \frac{s_i + t_i}{2}, \quad c_j = \frac{s_j + t_j}{2}.$$

If $c_i < c_j$, then e_i is expected to occur before e_j. For occurred events, we compute the observed centers using the annotated biases:

$$\hat{c}_i = \frac{(s_i + b_i^s) + (t_i + b_i^t)}{2}, \quad \hat{c}_j = \frac{(s_j + b_j^s) + (t_j + b_j^t)}{2}.$$

The pair is counted as correct only if both events occur and the observed ordering matches the requested ordering:

$$\mathbf{1}[o_i = 1, o_j = 1] \cdot \mathbf{1}[\mathrm{order}(\hat{c}_i, \hat{c}_j) = \mathrm{order}(c_i, c_j)].$$

If either event is missing, the pair is counted as incorrect. The final Order Accuracy is the fraction of correctly ordered event pairs:

$$\text{OrderAcc} = \frac{\sum_{i<j} \mathbf{1}[o_i = 1, o_j = 1] \cdot \mathbf{1}[\text{order}(\hat{c}_i, \hat{c}_j) = \text{order}(c_i, c_j)]}{\binom{N}{2}}.$$

This metric evaluates global temporal structure. A video may contain many requested events but still fail the prompt if their sequence is incorrect.

Overlap Accuracy. Overlap Accuracy measures whether events that are specified to overlap or occur concurrently in the prompt remain overlapping in the generated video. We first identify event pairs that overlap in the prompt:

$$\mathcal{O} = \{(i, j) : \min(t_i, t_j) > \max(s_i, s_j)\}.$$

For each pair $(i, j) \in \mathcal{O}$, we compute the observed intervals from the human-annotated biases:

$$\hat{s}_i = s_i + b_i^s, \quad \hat{t}_i = t_i + b_i^t.$$

The pair is counted as correct only if both events occur and their observed intervals overlap:

$$\mathbf{1}[o_i = 1, o_j = 1] \cdot \mathbf{1}[\min(\hat{t}_i, \hat{t}_j) > \max(\hat{s}_i, \hat{s}_j)].$$

The final Overlap Accuracy is:

$$\text{OverlapAcc} = \frac{1}{|\mathcal{O}|} \sum_{(i,j)\in\mathcal{O}} \mathbf{1}[o_i = 1, o_j = 1] \cdot \mathbf{1}[\min(\hat{t}_i, \hat{t}_j) > \max(\hat{s}_i, \hat{s}_j)].$$

If either event in an overlapping pair is missing, the pair is counted as incorrect. This metric is complementary to Order Accuracy: Order Accuracy evaluates before/after relations, while Overlap Accuracy evaluates concurrency preservation.

Temporal Constraint Satisfaction Rate. Temporal Constraint Satisfaction Rate (TCSR) summarizes whether temporal constraints are satisfied. This metric evaluates complete prompt-level temporal faithfulness as:

$$\text{TCSR} = \frac{1}{|\mathcal{C}|} \sum_{c\in\mathcal{C}} \mathbf{1}[c \text{ is satisfied}].$$

A constraint is satisfied only when the required event or event relation is realized and its temporal condition is met. Each event contributes two separately counted boundary constraints. Each requires the event to occur and its corresponding boundary deviation to lie within the tolerance, that is

- it is the start time of an interval constraint and the error from the ground truth start time is within 0.25s;
- or it is the end time of an interval constraint and the error from the ground truth end time is within 0.25s.

For order constraints, both events must occur and their before/after relation must be preserved. For overlap constraints, both events must occur and their observed intervals must overlap.

Rationale. These metrics are designed to jointly evaluate event realization, interval alignment, temporal ordering, concurrency, and full prompt-level constraint satisfaction. A key design choice is that missing events are explicitly penalized. Missing events reduce Event Occurrence, receive a duration-based penalty in Temporal Error, invalidate any related order or overlap pairs, and violate the corresponding TCSR constraints. This avoids a degenerate evaluation where a model appears temporally accurate only because it omits difficult events. By relying on human annotations of event occurrence and temporal deviations, the evaluation directly measures whether generated videos satisfy the intended prompt-level temporal structure.

D MORE RESULTS

D.1 RESULTS ON STORYBENCH

For StoryBench, we rescale event timelines to 10 seconds and use the background description as a persistent event spanning the full video. **TIE** outperforms previous multi-event video generation method on FVD, CLIP-Event, and VideoScore dimensions, indicating stronger temporal organization and visual fidelity.

Table A2: Text-to-Video Generation Result on StoryBench: The evaluation results of MinT come from their paper since they are not open-sourced.

Method	FID↓	FVD↓	CLIP-Event ↑	VQ↑	TC↑	DD↑	TA↑
MinT	40.87	484.44	0.270	2.56	2.44	3.32	2.92
RoTE	**23.58**	**236.45**	**0.290**	**3.39**	**3.21**	**3.57**	**3.14**

D.2 **TIE** AS PROMPT EXTENDER

A key advantage of **TIE** is its capacity for structured prompt extension. We use LLMs (e.g., Qwen3-Max) to expand simple VBench (Huang et al., 2024) prompts into our structured event representation. Compared to the official **WanExtender**, the **TIE**-enhanced variant shows superior fidelity to long-horizon descriptions. As shown in Table A3 and Figure A9, it generates more dynamic content that faithfully follows detailed spatial and temporal relationships, indicating that **TIE** provides a robust structured extension mechanism for complex video synthesis.

Table A3: Prompt Extending Results: These metrics are separated into four blocks related to Visual Quality, Motion, Consistency and Prompt Faithfulness. We find that WanExtender increases visual quality at the expense of dynamic degree, often producing nearly static scenes. In contrast, the **TIE**-enhanced variant significantly increases dynamic degree, retains comparable visual quality, and improves prompt faithfulness.

method	Dynamic Degree	Imaging Quality	Aesthetic Quality	Motion Smooth.	Human Action	Overall Consist.	Subject Consist.	Color	Object Class	Multiple Objects	Spatial Relation.	Tempo. Style	Appear. Style
Baseline	0.375	**0.665**	0.607	0.985	0.72	0.237	0.938	**0.895**	0.747	0.562	0.743	0.231	0.209
WanExtender	0.119	0.643	**0.658**	**0.993**	0.962	0.266	**0.948**	0.830	**0.925**	**0.797**	0.780	0.243	0.219
RoTE	**0.763**	0.623	0.599	0.979	**0.994**	**0.279**	0.924	0.892	0.899	0.741	**0.804**	**0.254**	**0.247**

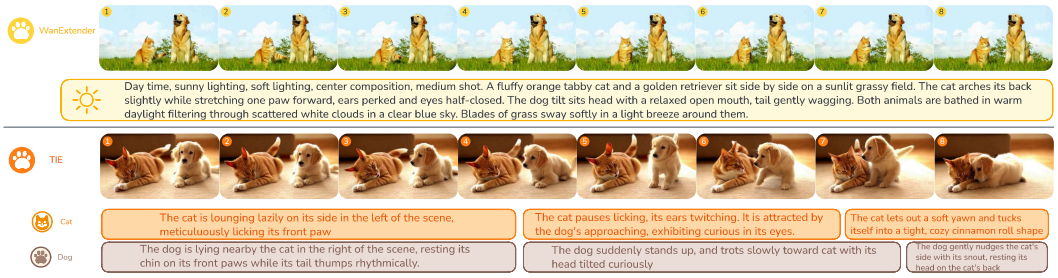

Figure A9: Comparison Results between WanExtender and the **TIE**-enhanced variant: the original short prompt from VBench is "a cat and a dog". VLM extends the short prompt into a series of consistent behaviors which result in an interaction between a dog and a cat.

D.3 CASE STUDY ON THE ROBOTICSEVENTS

In Figure A10, we show a case for precise temporal control of TIE. We adjust the time interval of one event of the right arm. The right arm should reach out to pull the door of the oven. We specify this accurate start time from $2.0s$ to $5.0s$, TIE accurately performs the requirement, while the Finetuned one does not respond in time.



Figure A10: Generation results on *RoboticsEvents*. We compare the temporal response behavior of the Finetuned baseline and **TIE**.

D.4 Boundary Experiments

Performance for different numbers of entities and events Evaluation results with different amount of descriptions split can be found in Table A4, and results with different amount of entities split can be found in Table A5. We can see that with more events and entities, the temporal alignment of **TIE** consistently drops, while more descriptions provide more information and increase the video quality.

Table A4: **TIE** Performance on various description count splits.

Description Count	CLIP-Event ↑	VQ ↑	TC ↑	DD ↑	TA ↑	FC ↑
[3,4]	0.2846	2.977	2.941	2.803	2.741	2.853
[5,6,7]	0.2694	2.975	2.920	2.907	2.768	2.851
[8,9,10]	0.2394	2.965	2.900	2.943	2.822	2.837
[11,12,13]	0.2267	3.057	2.987	3.072	2.948	2.934
[14,15,16]	0.2138	3.293	3.285	3.290	3.093	3.220

Table A5: **TIE** Performance on various entity count splits.

Entity Count	CLIP-Event ↑	VQ ↑	TC ↑	DD ↑	TA ↑	FC ↑
1	0.2801	2.942	2.887	2.829	2.711	2.811
2	0.2400	2.971	2.897	2.949	2.837	2.821
3	0.2221	3.238	3.212	3.209	3.047	3.161
4	0.2140	3.176	3.157	3.140	3.109	3.120
5	0.2150	3.310	3.338	3.299	3.143	3.289
6	0.1977	3.299	3.308	3.371	3.176	3.241
7	0.2089	3.501	3.542	3.658	3.223	3.503

D.5 Experiment Settings

The original experiments use 32 H800 GPUs for training. Additional experiments in Appendix F use separately reported settings. All 720p models (on Pexels-250k Dataset) are trained in batch size 512 for 6,500 steps with a learning rate of 10^{-5} and an AdamW (Kingma & Ba, 2015; Loshchilov & Hutter, 2019) optimizer. All inferences are performed with 50 steps, with a CFG scale of 5.0. All 480p models (on domain-specific datasets) are trained in batch size 128 for 20,000 steps with a learning rate of 3×10^{-5}.

E Discussions

E.1 Limitations

TIE encodes event intervals but does not explicitly enforce event–event relations or physical consistency. Uniform averaging can smooth sharp onsets, offsets, and internal event phases. The robustness bound requires non-degenerate normalization, and the experiments do not establish stability at arbitrary interval lengths or quality improvements across all backbones. The additional experiments report point estimates without quantifying run-to-run variability.

Modern VLMs still may not be capable enough to generate such a heavy prompt. Since captioning events requires understanding high-FPS videos and long contexts. VLM could generally output hallucinated contents, mismatched entities and behaviors, inconsistent entities and many other kinds of factual failures. Reflective Agent can somehow refine these problems but for modern VLMs that require API calls, retrying for too many times could be expensive.

E.2 Future Work

We believe **TIE** can be naturally integrated with long-video generation paradigms, such as next-chunk generation with KV-Cache, resulting in the next-level AIGC visual content creation suite. And also there could be natural applications for data generation / enrichment like the embodiment-AI scenario and digital human. We are on the way to explore the new frontier of interactive, customized and content-rich visual immersion / creation.

F ADDITIONAL EXPERIMENTAL RESULTS

This section presents additional results on interval conditioning, temporal kernels, normalization, duration, perturbed schedules, backbone transfer, and computational cost. We report point estimates and compare methods within each table's evaluation setting.

F.1 INTERVAL-AWARE CONDITIONING

Table A6 compares timestamp finetuning, center–length encoding, learned start/end embeddings, temporal bias, hard masking, and RoTE. Center–length encoding is the strongest alternative on nDTW, EMD, and FVD. RoTE improves nDTW from 0.2690 to 0.2805, reduces EMD from 0.1356 to 0.1304, and reduces FVD from 125.89 to 116.64. CLIP-Event decreases from 0.2115 to 0.2091, indicating a tradeoff between temporal alignment and this semantic metric.

F.2 ALTERNATIVE TEMPORAL KERNELS

Table A7 compares uniform integration with Quadratic weighting, weighted start–end features, endpoint averaging, and numerical averaging. Quadratic denotes the normalized upward-opening quadratic that emphasizes interval boundaries. Endpoint averaging differs from the original DoTE ablation, which concatenates separate start/end channel groups. RoTE achieves the best nDTW, EMD, and FVD in this comparison; the alternatives obtain higher CLIP-Event, and endpoint averaging obtains higher CLIP-Video.

F.3 NORMALIZATION

Self-overlap calibration improves FVD from 135.32 without normalization and 128.16 with L_2 normalization to 116.64 (Table A8). It also improves nDTW and EMD, while the unnormalized variant achieves the highest CLIP-Event. The calibration compensates for attenuation from averaging rotations; the probability kernel itself already has unit mass.

F.4 DURATION BUCKETS AND PERTURBED SCHEDULES

Table A9 reports CLIP margin and nDTW across event-duration buckets. RoTE improves both metrics in every bucket. The largest-duration bucket has smaller improvements and often contains slowly changing background descriptions. These results support comparisons within each bucket rather than rankings across different duration ranges.

Table A10 compares the original evaluation with perturbed schedules. TIE retains 90.08% TCSR with temporal error 0.127s, compared with 96.03% and 0.073s on the original evaluation. The degradation indicates sensitivity to schedule changes despite retained temporal control. The table does not provide a baseline evaluated on the same perturbed schedules, so it does not establish a paired OOD advantage or controlled short-to-long extrapolation.

Table A6: Interval-aware conditioning at 480p (8 H100 GPUs, 20,000 training steps). CLIP-E and CLIP-V denote CLIP-Event and CLIP-Video.

Method	nDTW ↑	EMD ↓	FVD ↓	CLIP-E ↑	CLIP-V ↑
Timestamp finetuning	0.2307	0.1533	145.20	0.2112	0.8432
Center–length	0.2690	0.1356	125.89	0.2115	0.8627
Learned start/end	0.2323	0.1527	167.50	0.2131	0.8431
Temporal bias ($\lambda = 4$)	0.2279	0.1531	179.06	0.2097	0.8457
Temporal bias ($\lambda = 0.5$)	0.2359	0.1526	139.72	0.2094	0.8467
Temporal mask	0.2193	0.1557	195.73	0.2107	0.8428
RoTE / sinc	0.2805	0.1304	116.64	0.2091	0.8650

Table A7: Alternative kernels at 480p (8 H100 GPUs, 20,000 training steps). Quadratic uses a normalized, endpoint-weighted quadratic kernel.

Method	nDTW ↑	EMD ↓	FVD ↓	CLIP-E ↑	CLIP-V ↑
RoTE / sinc	0.2805	0.1304	116.64	0.2091	0.8650
Quadratic	0.2707	0.1349	127.80	0.2126	0.8639
Weighted start–end	0.2705	0.1340	137.73	0.2129	0.8627
Start–end average	0.2782	0.1323	132.83	0.2120	0.8666
Numerical average ($N = 8$)	0.2749	0.1326	135.00	0.2131	0.8645

Table A8: Normalization ablation. The self-overlap calibration improves nDTW, EMD, and FVD over the alternatives.

Method	nDTW ↑	EMD ↓	FVD ↓	CLIP-E ↑	CLIP-V ↑
No normalization	0.2758	0.1335	135.32	0.2133	0.8648
L_2 normalization	0.2758	0.1319	128.16	0.2076	0.8647
Self-overlap (original)	0.2805	0.1304	116.64	0.2091	0.8650

Table A9: Duration-bucket results on 720p Pexels I2V. Differences are RoTE minus baseline; nDTW values are compared within each bucket, not across buckets.

Duration	RoTE margin	Base margin	Δ margin	RoTE nDTW	Base nDTW	Δ nDTW
0–1s	0.01142	0.00569	+0.00573	0.08785	0.08568	+0.00217
1–2s	0.00785	0.00502	+0.00283	0.04369	0.04319	+0.00050
2–3s	0.00988	0.00576	+0.00412	0.02578	0.02534	+0.00044
3–5s	0.01120	0.00749	+0.00371	0.01240	0.01184	+0.00056
5–10s	0.00381	0.00287	+0.00094	0.00324	0.00318	+0.00006

Table A10: Human-verified temporal metrics. All entries except temporal error (seconds) are percentages. Baseline and TIE use the original evaluation; TIE OOD uses perturbed schedules, so these are not matched OOD baseline comparisons.

Setting	Occurrence ↑	Error ↓	Order ↑	Overlap ↑	TCSR ↑
Baseline	80.45	0.261	63.37	66.23	77.34
TIE	96.03	0.073	92.32	88.64	96.03
TIE OOD	91.22	0.127	82.83	78.90	90.08

Table A11: Additional backbone comparison. RoTE improves temporal metrics on both backbones, but FVD worsens on Wan2.1.

Backbone / method	nDTW ↑	EMD ↓	FVD ↓	CLIP-E ↑	CLIP-V ↑
Wan2.2 TI2V-5B / base	0.2307	0.1533	145.20	0.2112	0.8432
Wan2.2 TI2V-5B / TIE	0.2805	0.1304	116.64	0.2091	0.8650
Wan2.1 T2V-1.3B / base	0.1566	0.1876	176.59	0.2061	0.8081
Wan2.1 T2V-1.3B / TIE	0.2080	0.1586	182.29	0.2106	0.8374

Table A12: Training cost averaged over stable iterations on 8 H100 GPUs. Memory entries are rounded; deltas use finer precision. A small negative timing difference does not establish a consistent speedup.

Setting	Step (s)	Peak (GB)	Time overhead	Memory delta
480p / base	5.425	53.3	–	–
480p / PyTorch	5.616	53.9	+3.5%	+0.59 GB
480p / Triton	5.518	53.3	+1.7%	≈ 0 GB
720p / base	15.912	62.2	–	–
720p / PyTorch	16.723	63.4	+5.1%	+1.20 GB
720p / Triton	15.669	62.2	-1.5%	≈ 0 GB

Table A13: Single-H100 inference, 50 denoising steps and 161 frames. Times are approximate seconds per denoising iteration. Rounding can cause small differences between displayed times and overhead percentages.

Setting	480p (s/it)	Overhead	720p (s/it)	Overhead
Base	1.35	–	4.45	–
TIE / PyTorch	1.38	+2.2%	4.53	+1.8%
TIE / Triton	1.37	+1.5%	4.46	+0.2%

F.5 Additional backbone

On Wan2.1 T2V-1.3B, nDTW improves from 0.1566 to 0.2080 and EMD from 0.1876 to 0.1586, while FVD worsens from 176.59 to 182.29 (Table A11). Temporal alignment improves on both tested backbones, but visual-quality gains do not transfer uniformly across models.

F.6 Training and inference cost

Table A12 reports training time and peak memory. The PyTorch implementation adds 3.5%/5.1% step time at 480p/720p. Triton adds 1.7% at 480p, while its -1.5% timing difference at 720p should not be interpreted as a consistent speedup. Triton has approximately unchanged peak memory; PyTorch increases memory by 0.59/1.20 GB.

Table A13 reports time per denoising iteration, rather than end-to-end latency. Triton adds approximately 1.5%/0.2% time at 480p/720p, compared with 2.2%/1.8% for PyTorch. Peak memory is 22.28 GB for the base and Triton implementations; PyTorch uses 22.28/22.37 GB at 480p/720p. These results support small implementation-dependent overhead in the tested workloads, rather than zero runtime cost.