TIMING IS NOT ENOUGH: MITIGATING CONCURRENT ACTION OMISSION IN VIDEO GENERATION

Anonymous authors
Paper under double-blind review

ABSTRACT

A prompt for video generation describes actions over time, and a subject is often asked to perform several of them at once. Temporal binding methods decide when each event appears and have made sequential multi-event generation more reliable. Once events contain concurrent actions, however, many requested actions still go missing, and those that appear are often not sustained. We trace the missing actions to two places along the conditioning pathway. In the text encoder, the subject that every frame must read mixes all its events, and isolating events from each other removes the actions that the subject carries. In video-text attention, the actions requested of one subject at the same time compete for the attention at its location, and this attention shifts from one action to another across frames. We propose **CoRAct**, a training-free approach that keeps the original prompt intact. It anchors each subject to the actions of its current event in the text representation and supports its co-required actions together at every frame in video-text attention. We further construct a benchmark of 84 prompts that specifies an interval for every action. Our protocol judges who performs each action and for how long, and it ranks the compared methods in the same order as human annotators. Across three backbones in image-to-video and text-to-video generation, CoRAct keeps the overall omission rate on par with existing training-free methods while significantly reducing the actions missing from their interval.

1 INTRODUCTION

A prompt for video generation describes a scene that unfolds over time. Unlike an image caption, which describes a single moment, it specifies which actions occur, which of them occur together, and in what order. Video diffusion transformers (Wan et al., 2025; Yang et al., 2025b; Kong et al., 2024) have made substantial progress in visual fidelity and motion coherence, yet they condition on such a prompt as image models do. Their text encoders contextualize the entire description in a single pass (Raffel et al., 2020; Wan et al., 2025; Yang et al., 2025b; Kong et al., 2024), and every frame reads the resulting representation regardless of the moment it depicts. In the prompt of Fig. 1, for instance, the man in a suit should keep talking and lifting sheet music while giving a high five, and the man wearing glasses should remove his glasses while returning it. Each subject thus passes through several *events*, the sets of actions requested of it at the same time, and for a moment both subjects must perform two or three co-required actions together. The generated videos often fail within these events, dropping the sheet music or giving the removal of the glasses to the wrong man. In both cases an action is not realized by its designated subject at the time at which it was requested together with the others. We refer to this failure as *concurrent action omission*, in which an action is missing, performed by another subject, or only partially performed at its requested time. We trace the missing actions to conditioning that overlooks the temporal structure of the prompt.

Recent methods bind text descriptions to time intervals, either through post-training with timestamped captions (Wu et al., 2025; Shu et al., 2026; Girish et al., 2025) or through inference-time attention modulation (Zhang et al., 2026; Xu et al., 2026; Chen et al., 2026), and have made sequential multi-event generation considerably more reliable. The inference-time methods decide which event of a jointly encoded prompt each frame reads, so the representation of that event is still formed from the whole prompt. Moreover, assigning an event to its interval specifies when its actions should occur, but not that its co-required actions occur together on the designated subject. The remaining challenge is compositional, and it arises often, since most real video clips contain events that overlap

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.

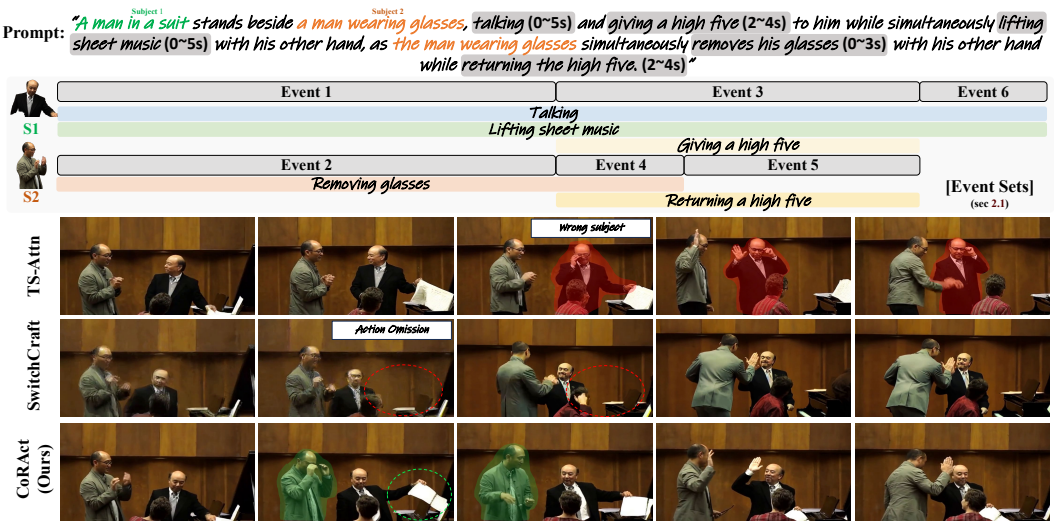


Figure 1: **Concurrent action omission.** Events 1 to 6 mark where the co-required actions of each subject change. Under temporal binding, an action is performed by the wrong subject (TS-Attn) or never appears in its interval (SwitchCraft), whereas CoRAct realizes the co-required actions on their designated subjects.

in time (Shu et al., 2026). Beyond composing multiple objects (Tian et al., 2024) or binding distinct actions to different subjects (Sun et al., 2025), the same subject should realize the co-required actions of an event concurrently. Yet it remains unclear whether temporal binding carries over to events with concurrent actions.

In this work, we find that temporal binding reduces omission only modestly once events contain concurrent actions. Many co-required actions still go missing at the time they are requested, and even those that appear are often not sustained over their interval. To locate why actions go missing, we analyze how the prompt is encoded and how the video attends to it while keeping temporal binding fixed. First, the text encoder mixes the representation of an event with the other events before temporal binding is applied, an effect we term *inter-event interference* (Sec. 2.3). The mention that introduces a subject attends to the verbs of its other events as much as to those of its current event, while the subject remains visible at every frame. A naive remedy that cuts attention between events inside the encoder also removes the actions that the subject carries and leaves more actions missing. Second, even without other events, the actions requested of one subject at the same time compete for the attention of the video tokens at the location of that subject, an effect we term *intra-event action competition* (Sec. 2.4). This attention shifts from one action to another across frames, and an action left without it at a frame is not performed there. These two causes call for two requirements, anchoring each subject to the actions of its current event without losing the actions it carries, and supporting the co-required actions of an event together at the location of their subject at every frame of its interval.

Guided by these requirements, we propose **CoRAct (Co-Required Action realization)**, a training-free approach that binds each subject to the actions of its current event at both stages of conditioning. In the text representation, it derives a representation for each interval from the jointly encoded prompt. Each subject representation is rotated toward the actions of its current event, so that it is anchored to its own event while keeping the shared context. In video-text attention, CoRAct guides the latent so that every co-required action receives attention at the location of its subject at every frame. Unlike attention guidance for images (Chefer et al., 2023) and attention routing between events (Zhang et al., 2026), the proposed objective balances the co-required actions of one event against each other, so that neither attention shifting from one action to another nor an action on another subject satisfies it.

We also construct a benchmark and an evaluation protocol for concurrent actions. Unlike existing benchmarks for consecutive events (Wang et al., 2025) or action-subject binding (Sun et al., 2025), our benchmark specifies an interval for every action and comprises 84 prompts, 54 with up to six co-required actions of up to three subjects and 30 with consecutive events. Our protocol counts the

actions that the designated subject does not perform over most of their interval, and it ranks the compared methods in the same order as human annotators. Across three backbones, CoRAct keeps the overall omission rate on par with existing training-free methods while significantly reducing the actions missing from their interval.

2 Analysis of Concurrent Action Omission

This section examines what temporal binding leaves behind once events contain concurrent actions and where the remaining failures arise. We first formalize events and the success criteria (Sec. 2.1), then measure the effect of temporal binding (Sec. 2.2), and locate two causes, one in the text representation (Sec. 2.3) and one in video-text attention (Sec. 2.4). All analyses use the unmodified backbone with shared prompts, initial noise, and sampling settings (Appendix A).

2.1 Problem Setup

A prompt for video generation specifies a set of actions, each performed by a subject, possibly on an object, over a time interval. Several actions of the same subject may be requested over the same period, in which case they should be performed concurrently, and the set of actions requested for that subject changes only when one of them begins or ends. We therefore group the actions of each subject into events, as introduced in Sec. 1.

We define an event as $e_k = (s_k, \mathcal{A}_k, \tau_k)$, consisting of a subject s_k, the set $\mathcal{A}_k$ of actions requested of it simultaneously, and a maximal interval τ_k over which this set does not change, a formulation inspired by person-centric and temporally dense action annotation (Gu et al., 2018; Yeung et al., 2018). The actions in $\mathcal{A}_k$ are therefore co-required throughout τ_k, an action that spans several events belongs to each of them, and events of different subjects may share time. As the default binding mechanism we use interval masking, a hard form of the attention restriction used in prior work (Wu et al., 2025; Chen et al., 2026). At each frame, it hides the verb phrases of the actions that are not requested at that time, while every frame still reads the rest of the prompt, including every mention of the subjects.

In addition, we measure omission at the level of actions. An action is omitted when its designated subject does not perform it over most of its interval. We call an omitted action *absent* when it does not appear in its interval and *partial* when it appears but covers only part of the interval, and we count the actions performed by another subject separately. The omission rate is the fraction of omitted actions, following the protocol in Sec. 4.1. The order of events is measured by the StoryEval protocol (Wang et al., 2025). Our analysis uses the two splits of our benchmark, a mixed split whose events contain concurrent actions and a sequential split of consecutive events, together with 20 single-event prompts derived from the mixed split (Appendix B).

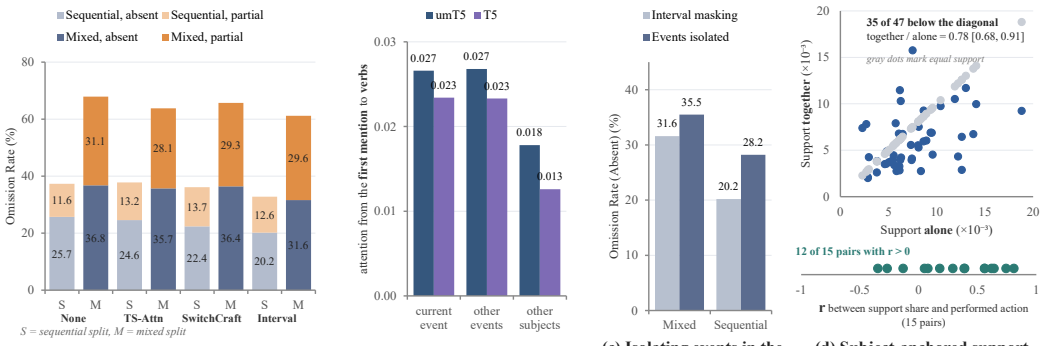
2.2 Temporal Binding on Concurrent Actions

Temporally controlled generation assumes that binding text to time is enough to realize the requested actions (Wu et al., 2025). Binding decides which event each frame reads, so it should keep an action from going missing at the time it is requested. We test this by comparing the unmodified backbone with TS-Attn (Zhang et al., 2026), SwitchCraft (Xu et al., 2026), and interval masking on the mixed split, where events contain concurrent actions, and on the sequential split (Table 1). On the sequential split, where most events contain a single action, these methods omit 35 to 40% of the requested actions, whereas on the mixed split they omit 62 to 69%, and binding narrows this gap only slightly (Fig. 2(a)).

Table 1: Temporal binding on Wan2.2 TI2V-5B. Omission rate (%) with absent (Abs.) and partial (Part.) actions, and the order of events.

	Mixed				Sequential	
Binding	Om.↓	Abs.	Part.	Ord.↑	Om.↓	Ord.↑
None	68.9	36.8	31.1	.178	38.7	.256
TS-Attn	64.3	35.7	28.1	.195	39.5	.231
SwitchCraft	66.6	36.4	29.3	.150	37.1	.292
Interval masking	61.6	31.6	29.6	.170	34.8	.308
+ events cut	68.8	35.5	32.1	.077	39.9	.330

On the mixed split, binding lowers the omission rate significantly only under interval masking, which also reduces absent actions by a few points, whereas no method reduces partial actions significantly



(a) Omission rate under temporal binding (b) First-mention attention to verbs (c) Isolating events in the text encoder (d) Subject-anchored support, alone vs. together

Figure 2: Analysis on Wan2.2 TI2V-5B. (a) Absent and partial actions on the sequential (S) and mixed (M) splits. (b) First-mention attention to the verbs of each event and of other subjects. (c) Absent actions when events are isolated in the text encoder. (d) Support of each action requested alone and together, with gray dots marking equal support, and the correlation between support share and performed action over 15 pairs (bottom).

on either split (Appendix E). Since an action has to appear before it can be sustained, we ask why actions go missing. The frames of an interval no longer read the actions of other events, so we first examine whether the representation of the event is itself affected by the other events (Sec. 2.3). After that, we explore how the video tokens divide their attention among the co-required actions of an event, which remain readable at every frame (Sec. 2.4).

2.3 Inter-event Interference in the Text Representation

Interval masking restricts which tokens each frame reads, but the tokens it lets through are produced by encoding the whole prompt at once. We therefore examine whether the representation of an event already carries the other events, using the text encoders of our backbones, umT5 (Chung et al., 2023) in Wan2.2 and T5 (Raffel et al., 2020) in CogVideoX. When the actions of other events are present, the representations of the actions of a target event move further from their representations with those actions hidden than when the same positions hold neutral words, with median distances of 0.233 against 0.127 for umT5 and 0.187 against 0.138 for T5 (Appendix F). The other events thus alter an event beyond the effect of added text.

This matters most for the subject. Every frame must read the subject to know who acts, so interval masking never hides it. In both encoders, the first mention of a subject attends to the verbs of its other events as much as to those of its current event, with ratios of 1.01 for umT5 and 1.00 for T5, although it attends much less to the verbs of other subjects (Fig. 2(b)). The encoder thus separates who acts but not when. Every frame therefore reads a subject that mixes all its events, and binding cannot remove this mixture (Appendix F). This is the *inter-event interference* introduced in Sec. 1.

A naive remedy is to isolate events from each other inside the encoder. When the shared description, including the subject, can no longer attend to any action, absent actions rise from 31.6% to 35.5% on the mixed split and from 20.2% to 28.2% on the sequential split (Table 1, Fig. 2(c)). When the shared description may still attend to every action, absent actions change less, yet omission still does not fall (Appendix E). Isolating events thus removes the interference together with the actions that the subject carries. The subject should instead stay bound to its actions while being anchored to those of its current event.

2.4 Intra-event Action Competition in Video-text Attention

The single-event prompts each describe one event, in which one subject performs several co-required actions at the same time. Since there is no other event, inter-event interference cannot arise, and every frame reads all the actions. Even so, 45% of the requested actions are omitted (Appendix G), and we examine how the video tokens divide their attention among co-required actions.

For a co-required action a of subject s_a at latent frame f, let $V_{a,f}(x)$ be the cross-attention from location x to the verb tokens of a and $\hat{S}_{s,f}(x)$ the attention to the subject normalized to a maximum of one within the frame. We measure the subject-anchored support $q_{a,f} = \max_x \hat{S}_{s_a,f}(x)\, V_{a,f}(x)$, which is large only when the action is attended to where its subject is. An action requested together with others retains only 0.78 of the support it receives alone (Fig. 2(d)). The co-required actions then divide the remaining support from frame to frame, and the share held by each action tracks whether it is performed at that frame (Appendix G). An action left with little support at a frame is thus not performed there. This is the *intra-event action competition* introduced in Sec. 1.

The support also decides the outcome. When the attention to an omitted action is raised during the early denoising steps, the omission rate on the 12 failed prompts falls from 65.3% to 52.8% (Appendix G). A single raise, however, favors one action, whereas the competition involves all co-required actions at every frame. The co-required actions of an event should therefore be supported together at the location of their subject at every frame of its interval.

3 Method

The analysis in Sec. 2 points to two requirements. The subject of each event should be anchored to the actions of its current event without losing the actions it carries, and the co-required actions of an event should be supported together at the location of their subject at every frame of its interval. CoRAct meets the first in the text representation (Sec. 3.1) and the second in video-text attention (Sec. 3.2), without retraining the backbone or rewriting the prompt (Fig. 3).

3.1 Event-anchored Subject Representation

The subject as the anchor. Let $E \in \mathbb{R}^{L\times D}$ be the input embeddings of the L prompt tokens and $F(\cdot)$ the text encoder, so that $H = F(E)$ holds one row per token. Since the subject read by every frame mixes all its events (Sec. 2.3), we anchor its representation to the actions of its current event. Each event e_k has one subject s_k, while a subject may have several events. For event e_k, let p_k be the position of the mention through which its actions refer to s_k, namely the head noun of the subject phrase or a pronoun, and let $h_k = H_{p_k}$ be its row. We denote by $\mathcal{T}_{k,a}$ the positions of the verb and object tokens of action $a \in \mathcal{A}_k$. When a prompt contains several subjects, we first orthogonalize the rows of different subjects (Seo et al., 2025), so that anchoring one subject to its actions does not pull another subject toward them, and denote the result by $\bar{h}_k$ with direction $\hat{h}_k = \bar{h}_k / \|\bar{h}_k\|$.

Action direction from the text encoder. Isolating events inside the encoder strips the subject of its actions (Sec. 2.3), so we derive the direction of each action within the jointly encoded prompt. For each action a separately, we place the input embeddings of its verb and object tokens on the input row of its subject as a virtual perturbation and read the response of the encoder on the same row through a Jacobian-vector product, which is the directional derivative of F at E,

$$c_{k,a} = \left[J_F(E)\,\Delta_{k,a}\right]_{p_k}, \qquad (\Delta_{k,a})_i = \begin{cases} \sum_{j\in\mathcal{T}_{k,a}} E_j, & i = p_k, \\ 0, & i \neq p_k. \end{cases} \tag{1}$$

Evaluated at the original prompt, the response passes through the contextualization of the full prompt, so it turns the subject toward its action without discarding the context that the prompt shares across events. Unlike token merging for images (Hu et al., 2024), which sums attribute embeddings into an object token once, this direction follows the response of the encoder and differs from one event to the next. We keep only the component orthogonal to the subject, $d_{k,a} = c_{k,a} - \langle c_{k,a}, \hat{h}_k\rangle\, \hat{h}_k$, and bound its magnitude so that a weak response is not amplified,

$$b_{k,a} = \frac{d_{k,a}}{\sqrt{\|d_{k,a}\|^2 + (\rho\|\bar{h}_k\|)^2}}, \tag{2}$$

so that every $b_{k,a}$ is orthogonal to $\hat{h}_k$ and shorter than one. Neither the action tokens nor the subject are changed at this point, since $b_{k,a}$ only indicates the direction in which action a would move the subject.

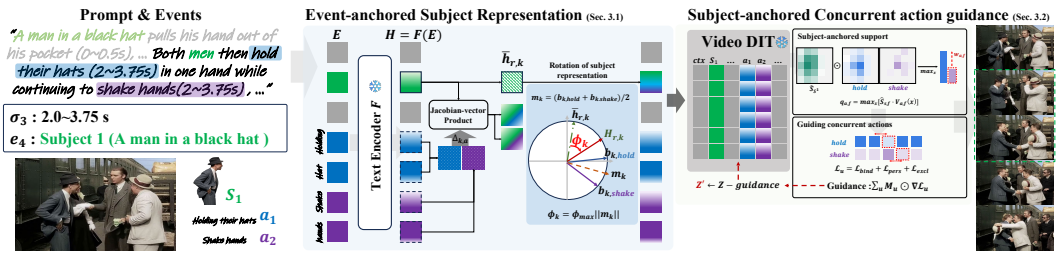


Figure 3: **Overview of CoRAct.** In the segment σ_3, the man in the black hat holds his hat and shakes hands (e_4). The text-side component rotates his subject representation toward the directions of these two actions, obtained from the frozen text encoder by a Jacobian-vector product. The attention-side component guides the latent so that both actions receive support at his location at every frame of σ_3. The prompt is abbreviated for display, and the full prompt is encoded at once.

Rotation toward the current event. Since the events of different subjects begin and end at different times, we partition the latent frames into segments $\{\sigma_u\}$ at the boundaries of all intervals, so that every subject stays in one event within a segment. The directions of the co-required actions of event e_k are averaged into a single direction $m_k = |\mathcal{A}_k|^{-1} \sum_{a \in \mathcal{A}_k} b_{k,a}$ rather than applied to separate rows, since they already compete for attention at the location of their subject (Sec. 2.4). Since every direction is bounded before averaging, no single action dominates m_k, the length of m_k grows with the agreement of the directions, and m_k remains orthogonal to $\hat{h}_k$. Starting from a fresh copy of H in every segment, so that no segment inherits the actions of another, we rotate the subject of every event whose interval contains σ_u toward m_k while preserving its norm,

$$H^{(u)}_{p_k} = \|\bar{h}_k\| \Big(\cos\phi_k \, \hat{h}_k + \sin\phi_k \, \frac{m_k}{\|m_k\|} \Big), \qquad \phi_k = \phi_{\max} \|m_k\|. \tag{3}$$

Actions pulling in a common direction thus rotate the subject by up to $\phi_{\max}$, while disagreeing directions leave it close to the original. The frames of σ_u attend to $H^{(u)}$ with the verb phrases of inactive actions hidden as in interval masking, so that each frame reads a subject turned toward the actions of its current event.

3.2 Subject-anchored Concurrent Action Guidance

Sec. 2.4 shows that the subject-anchored support shifts between co-required actions from frame to frame, and that raising the support of one omitted action lowers omission while favoring that action at one moment. We therefore raise the support of all co-required actions together at every frame and ask it to persist.

Subject-anchored support. Let $\mathcal{K}_u$ be the set of events whose interval contains σ_u. During sampling we compute the subject-anchored support $q_{a,f}$ of Sec. 2.4 for every action $a \in \mathcal{A}_k$ with $k \in \mathcal{K}_u$ and every latent frame $f \in \sigma_u$, where the subject of a is s_k. The verb and subject maps are averaged over tokens, heads, and the guided blocks and smoothed spatially. The subject map includes the rows rotated in Sec. 3.1 and is detached, so that guidance moves the action toward the subject rather than the subject toward the action.

Guiding concurrent actions. For segment σ_u, we raise the support of the co-required actions of every active event at every frame,

$$\mathcal{L}_{\text{bind}} = \sum_{k \in \mathcal{K}_u} \frac{1}{|\sigma_u|} \sum_{f \in \sigma_u} \sum_{a \in \mathcal{A}_k} w_{a,f} \big(-\log(q_{a,f} + \epsilon) \big), \qquad w_{a,f} = \underset{a \in \mathcal{A}_k}{\text{softmax}}\big(-q_{a,f} / T_f \big), \tag{4}$$

where the weights are detached and the temperature T_f is proportional to the mean support at frame f. Since the objective is evaluated at every frame, an action that holds the support of its subject in only some frames keeps the loss high, so support shifting between actions is penalized. The soft minimum puts most weight on the action with the least support at each frame. Since the competition recurs at every frame, we also penalize the change of support between neighboring frames,

$$\mathcal{L}_{\text{pers}} = \sum_{k \in \mathcal{K}_u} \sum_{a \in \mathcal{A}_k} \sum_{f,\, f+1 \in \sigma_u} \big(\log(q_{a,f} + \epsilon) - \log(q_{a,f+1} + \epsilon) \big)^2. \tag{5}$$

To keep an action on its own subject when a prompt has several subjects, we suppress its attention at the location of every other subject, including subjects without an action in the segment, relative to its own support,

$$\mathcal{L}_{\text{excl}} = \sum_{k \in \mathcal{K}_u} \sum_{a \in \mathcal{A}_k} \sum_{s' \neq s_k} \frac{1}{|\sigma_u|} \sum_{f \in \sigma_u} \frac{\max_x \hat{S}_{s',f}(x)\, V_{a,f}(x)}{q_{a,f} + \epsilon}. \tag{6}$$

With the loss of each segment $\mathcal{L}_u = \mathcal{L}_{\text{bind}} + \lambda_p \mathcal{L}_{\text{pers}} + \lambda_e \mathcal{L}_{\text{excl}}$, we update the latent z_t after each denoising step t in a set $\mathcal{S}$ of early steps as follows,

$$z_t \leftarrow z_t - \eta_t \sum_u M_u \odot \nabla_{z_t} \mathcal{L}_u, \tag{7}$$

where M_u selects the latent frames of σ_u, so that the guidance of one event does not alter the frames of another. Attention guidance for images recovers a neglected concept in a single image (Chefer et al., 2023), and TS-Attn strengthens all actions of an event together (Zhang et al., 2026), whereas our objective balances the co-required actions of one event against each other at every frame and only where their subject is. The set $\mathcal{S}$ and the other hyperparameters are given in Appendix I.

4 Experiments

4.1 Setup

Benchmarks. We evaluate on the 84 prompts of our benchmark, 54 in the mixed split and 30 in the sequential split collected from the results of MinT (Wu et al., 2025) (Appendix A).

Backbones and baselines. We apply CoRAct to CogVideoX1.5-5B (Yang et al., 2025b), Wan2.2 TI2V-5B, and Wan2.2 A14B (Wan et al., 2025) in both image-to-video and text-to-video generation, and compare it with the backbones and with the training-free methods TS-Attn (Zhang et al., 2026) and SwitchCraft (Xu et al., 2026). TIE (Shu et al., 2026), which is trained with event timestamps, is available for Wan2.2 TI2V-5B only, and MinT (Wu et al., 2025) is not included since its weights are not public. All methods share the prompt, initial noise, and sampling settings of each backbone (Appendix A).

Evaluation. Existing protocols judge whether consecutive events are completed (Wang et al., 2025) or whether a clip matches its caption (Wu et al., 2025), without checking who performs each action and for how long. StoryEval, for instance, judges each event as a whole from a description of keyframes, so it cannot tell whether the concurrent actions of an event each appear in their intervals (Appendix D). We therefore judge every action within its own interval with Qwen3.8-27B (Yang et al., 2025a), which receives the frames of the interval and the first frame with each designated subject outlined, and reports whether the action is performed, by whom, and over which frames (Appendix C). An action is realized when the designated subject performs it over at least $\theta = 0.7$ of its interval, so that two realized co-required actions overlap for at least 40% of their interval. We report the omission rate (OM), the fraction of requested actions that are not realized averaged over prompts, together with OM_P, which counts only the actions that never appear, and OM_S, which adds those performed by another subject. The order of events follows StoryEval, and visual quality (VQ) is the mean of four quality dimensions of VBench (Zheng et al., 2025). To validate the judge, 9 annotators watched the videos of CoRAct, TS-Attn, SwitchCraft, and TIE on Wan2.2 TI2V-5B for 10 prompts of our benchmark and marked whether each event is performed as requested. The judge agrees with the annotators on 64.4% of the events, against 84.3% between annotators, and ranks CoRAct, SwitchCraft, and TS-Attn in the same order as the annotators. Two independent runs of the judge on the same videos agree on 92.8% of the actions (Appendix C).

4.2 Experimental Results

Quantitative results. Table 2 compares CoRAct with the baselines on all splits and backbones. On the mixed split, CoRAct has the lowest OM_P among the training-free methods on Wan2.2 in image-to-video generation and on Wan2.2 A14B in text-to-video generation, although differences in a single setting are often small relative to the variation over prompts. Under human judgment,

Table 2: Quantitative comparison on the two splits of our benchmark. OM is the omission rate. OM_P counts only the actions that never appear, and OM_S adds those performed by another subject. †TIE is trained with event timestamps and is shown separately from the training-free methods. **Bold** and underline mark the best and second best among the training-free methods.

	Image-to-video								Text-to-video							
	Ours (mixed)					Ours (seq.)			Ours (mixed)					Ours (seq.)		
Method	OM_P↓	OM_S↓	OM↓	Order↑	VQ↑	OM↓	Order↑	VQ↑	OM_P↓	OM_S↓	OM↓	Order↑	VQ↑	OM↓	Order↑	VQ↑
CogVideoX1.5-5B																
Backbone	47.2	47.7	70.6	.152	**.693**	57.1	.324	**.738**	52.3	53.8	76.9	.082	.644	62.0	.239	**.679**
TS-Attn	**43.6**	**44.0**	72.1	.171	.658	54.8	.366	.707	51.4	52.0	**75.3**	.097	**.647**	63.7	**.352**	.670
SwitchCraft	47.4	47.9	72.2	.093	.659	56.3	.261	.705	57.0	58.5	82.8	.101	.619	**61.6**	.272	.650
CoRAct	46.3	46.3	**69.5**	**.173**	.667	**51.2**	**.432**	.698	**48.0**	**49.5**	75.5	**.102**	.625	63.5	.206	.621
Wan2.2 TI2V-5B																
Backbone	36.8	37.8	68.9	.178	.705	43.1	.420	.725	38.6	39.0	69.4	**.222**	**.707**	49.1	.277	.699
TS-Attn	35.7	36.2	**64.3**	.195	**.712**	41.6	.414	**.726**	38.9	39.5	67.7	.199	**.707**	**45.0**	.304	**.702**
SwitchCraft	36.4	37.3	66.6	.150	.703	37.7	.405	.718	43.4	43.8	66.6	.158	.694	52.7	.183	.669
CoRAct	**28.7**	**29.2**	64.5	**.223**	.704	**36.2**	**.430**	.715	**37.6**	**38.6**	**63.7**	.148	.703	49.2	**.322**	.683
TIE†	23.2	24.1	64.1	.285	.691	51.2	.407	.693	31.3	32.1	72.2	.295	.652	58.4	.284	.628
Wan2.2 A14B																
Backbone	13.6	13.6	64.6	.428	.723	44.5	.502	**.762**	13.8	15.3	61.7	.378	**.763**	41.0	.514	**.769**
TS-Attn	14.6	14.8	65.7	.403	.713	44.7	.503	.760	14.7	16.1	60.9	.379	.760	**36.3**	.581	.762
SwitchCraft	12.1	12.3	61.5	.425	.718	40.0	**.520**	.761	8.3	9.2	57.2	.501	.761	39.9	.666	.765
CoRAct	**10.4**	**10.5**	**58.2**	**.432**	**.724**	**38.3**	.487	**.762**	**7.0**	**7.2**	**56.6**	**.523**	.756	41.6	**.669**	.755

Table 3: Differences pooled over the six settings, in points with bootstrap 95% intervals over prompts. Bold values exclude zero.

	OM_P	OM_S
CoRAct − Backbone	$\mathbf{-2.8}$ [−4.9, −0.6]	$\mathbf{-2.9}$ [−5.1, −0.5]
CoRAct − TS-Attn	$\mathbf{-2.2}$ [−4.3, −0.1]	−2.1 [−4.3, +0.1]
CoRAct − SwitchCraft	$\mathbf{-3.2}$ [−5.5, −0.8]	$\mathbf{-3.2}$ [−5.7, −0.7]
TS-Attn − Backbone	−0.6 [−2.5, +1.4]	−0.8 [−3.0, +1.3]
SwitchCraft − Backbone	+0.4 [−2.0, +2.8]	+0.3 [−2.2, +2.8]

Table 4: Ablation on Wan2.2 A14B, image-to-video, mixed split. Interval masking is applied in every row except the backbone.

	OM_P ↓	OM_S ↓	OM↓	Order↑
Backbone	17.0	18.1	57.5	.454
+ Text-side	12.9	12.5	51.6	.495
+ Attention-side (CoRAct)	12.3	12.9	51.1	.451

CoRAct has the highest rate of events performed as requested, 31.1% against 19.6%, 27.7%, and 5.4% for TS-Attn, SwitchCraft, and TIE, and the differences from TS-Attn and TIE are significant under a paired test (Appendix C).

Which omissions are reduced. Table 3 pools the differences over the six settings and shows where CoRAct acts. For the actions that never appear, the intervals of CoRAct lie entirely below zero against the backbone, TS-Attn, and SwitchCraft, whereas those of TS-Attn and SwitchCraft against the backbone include zero. The missing actions that Sec. 2 traced to inter-event interference and intra-event action competition are therefore reduced only by CoRAct. The actions that appear but fall short of their interval, the gap between OM_S and OM, do not decrease where OM_P does, which is why OM changes less than OM_P. Its OM is still significantly lower than that of the backbone, and with the looser threshold $\theta = 0.5$ it is significantly lower than that of every baseline (Appendix C).

Component ablation. Table 4 adds the two components in turn on Wan2.2 A14B. The text-side component is applied together with interval masking, so its row also contains the effect of binding text to time, and it lowers OM_P from 17.0 to 12.9 and the omission rate from 57.5 to 51.6 while raising the order of events. The attention-side component then lowers both further, to 12.3 and 51.1, in line with its motivation in Sec. 2.4, although the order of events returns to the level of the backbone. On Wan2.2 TI2V-5B, where its share is larger, the attention-side component lowers the actions that never appear by 4.7 points (95% interval $[-8.8, -0.6]$, Appendix H).

4.3 Qualitative Results

Fig. 4 compares the methods on two prompts. In image-to-video generation (left), CoRAct lets the woman cover her mouth, then turn back to the audience while waving, and finally bow. Each baseline omits one of these actions. SwitchCraft and TS-Attn keep her facing the camera instead of turning back, and TIE does not show her waving. In text-to-video generation (right), our method shows the



Figure 4: **Qualitative comparison** in image-to-video (left) and text-to-video (right) generation. Prompts are abbreviated for display, and the bars show the requested interval of each action. Only CoRAct performs every requested action of the woman on the left, and on the right it keeps the man kneeling while the woman takes out the ring, raises her hands, and turns around with the bouquet.

man kneeling down and keeps him on his knees while the woman takes out the ring, raises her hands, and turns around with the bouquet in her hand. TS-Attn keeps the man standing while he holds out the box, SwitchCraft drops the bouquet when the woman raises her hands, and TIE does not keep both subjects in view.

5 Related Work

Early multi-event methods generate one clip per event and connect consecutive clips (Oh et al., 2023; Cai et al., 2024), which can break the continuity of subjects and scenes. Recent methods generate the whole video in one pass and bind each part of the prompt to a time interval, either by fine-tuning with separately encoded event captions (Wu et al., 2025) or interval-level rotary encoding (Shu et al., 2026), or by modulating the cross-attention of a pretrained model without training (Zhang et al., 2026; Xu et al., 2026; Chen et al., 2026). These methods control when each event appears, whereas compositional video generation binds content across regions and time (Tian et al., 2024) or balances spatiotemporal attention flow (Yang & Wang, 2024) without tying concurrent actions to their subjects. In text-to-image generation, similar failures have been traced to cross-attention (Chefer et al., 2023; Rassin et al., 2023) and to the text embedding (Hu et al., 2024; Seo et al., 2025; Chen et al., 2024; Baek et al., 2026), where attributes are bound to entities once for a whole image, and 1Prompt1Story (Liu et al., 2025) suppresses the descriptions of other images within one prompt, a setting close to the interference between events that we study. We address the actions that one subject must perform at the same time, which persist as omissions under temporal binding, compete for the same subject, and change from one event to the next.

6 Conclusion

Video diffusion models read a prompt that unfolds over time much as image models read a caption. Binding text to time tells each frame which event to read, yet once events contain concurrent actions, many of the requested actions still go missing and many of those that appear are not sustained. We traced the missing actions to two places along the conditioning pathway. In the text encoder, the subject that every frame reads mixes all its events. In video-text attention, the actions requested of one subject at the same time compete for the attention at its location. CoRAct addresses both without training, anchoring each subject to the actions of its current event and supporting its co-required actions together at every frame. Across three backbones in image-to-video and text-to-video generation, it keeps the overall omission rate on par with existing training-free methods while being the only one that significantly reduces the actions missing from their interval. Sustaining an action over its entire interval once it appears remains open for every method we evaluate, and we hope that our benchmark and protocol, which separate these two failures, help future work address it.

AI USE STATEMENT

In this work, we used generative AI tools to organize and clean the benchmark annotations, to correct typos and phrasing in the paper, and to review our implementation of CoRAct for errors. We have not used generative AI tools to generate synthetic data sets, to formulate mathematical claims, or to write proofs, and translation is not applicable to this work. Separately, a vision-language model serves as the judge of our evaluation protocol, and its agreement with human annotators is reported in Sec. 4.1. The authors reviewed all AI-assisted work and checked every reported number against the raw judge outputs and the logs of the generation and analysis runs. We take responsibility for the final content of this work, including text, claims, and artifacts produced with the aid of generative AI.

ETHICS STATEMENT

All data in this work come from public datasets, and our benchmark is constructed from the prompts, frames, and annotations of public video datasets, which we use for research purposes only. The human validation asks annotators only to judge whether actions are performed in generated videos, and no personal data are collected. Improving the control of human actions in generated videos may make realistic fabricated videos of people easier to produce, although our method does not change the capability of the backbones to render people.

REPRODUCIBILITY STATEMENT

The event formulation and evaluation criteria are defined in Sec. 2.1 and Sec. 4.1, and CoRAct is specified in Sec. 3. The benchmark, the judge prompt and decoding settings, and all hyperparameters are given in Appendix A, Appendix C, and Appendix I. All compared conditions share the prompt, initial noise, and sampling settings with a fixed seed. We will release the code, the benchmark, and the evaluation scripts.

REFERENCES

Kanghyun Baek, Jaihyun Lew, Chaehun Shin, Jungbeom Lee, and Sungroh Yoon. Diagnosing and correcting concept omission in multimodal diffusion transformers. *ArXiv*, abs/2605.14270, 2026. URL https://api.semanticscholar.org/CorpusID:288253889.

Ming-Hong 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. *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 7763–7772, 2024. URL https://api.semanticscholar.org/CorpusID:274992610.

Hila Chefer, Yuval Alaluf, Yael Vinker, Lior Wolf, and Daniel Cohen-Or. Attend-and-excite: Attention-based semantic guidance for text-to-image diffusion models. *ACM Transactions on Graphics (TOG)*, 42:1 – 10, 2023. URL https://api.semanticscholar.org/CorpusID:256416326.

Chieh-Yun Chen, Chiang Tseng, Li-Wu Tsao, and Hong-Han Shuai. A cat is a cat (not a dog!): Unraveling information mix-ups in text-to-image encoders through causal analysis and embedding optimization. *ArXiv*, abs/2410.00321, 2024. URL https://api.semanticscholar.org/CorpusID:273022617.

Gordon Chen, Ziqi Huang, and Ziwei Liu. Prompt relay: Inference-time temporal control for multi-event video generation. *arXiv preprint arXiv:2604.10030*, 2026.

Hyung Won Chung, Xavier Garcia, Adam Roberts, Yi Tay, Orhan Firat, Sharan Narang, and Noah Constant. Unimax: Fairer and more effective language sampling for large-scale multilingual pretraining. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=kXwdL1cWOAi.

Sharath Girish, Viacheslav Ivanov, Tsai-Shien Chen, Hao Chen, Aliaksandr Siarohin, and Sergey Tulyakov. Alchemint: Fine-grained temporal control for multi-reference consistent video generation. *ArXiv*, abs/2512.10943, 2025. URL https://api.semanticscholar.org/CorpusID:283737729.

Chunhui Gu, Chen Sun, David A Ross, Carl Vondrick, Caroline Pantofaru, Yeqing Li, Sudheendra Vijayanarasimhan, George Toderici, Susanna Ricco, Rahul Sukthankar, et al. Ava: A video dataset of spatio-temporally localized atomic visual actions. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 6047–6056. IEEE, 2018.

Taihang Hu, Linxuan Li, Joost van de Weijer, Hongcheng Gao, Fahad Shahbaz Khan, Jian Yang, Ming-Ming Cheng, Kai Wang, and Yaxing Wang. Token merging for training-free semantic binding in text-to-image synthesis. *ArXiv*, abs/2411.07132, 2024. URL https://api.semanticscholar.org/CorpusID:273962963.

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

Tao Liu, Kai Wang, Senmao Li, Joost van de Weijer, Fahad Shahbaz Khan, Shiqi Yang, Yaxing Wang, Jian Yang, and Ming-Ming Cheng. One-prompt-one-story: Free-lunch consistent text-to-image generation using a single prompt. *ArXiv*, abs/2501.13554, 2025. URL https://api.semanticscholar.org/CorpusID:275820221.

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*, 2023. URL https://api.semanticscholar.org/CorpusID:266055534.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of machine learning research*, 21(140):1–67, 2020.

Royi Rassin, Eran Hirsch, Daniel Glickman, Shauli Ravfogel, Yoav Goldberg, and Gal Chechik. Linguistic binding in diffusion models: Enhancing attribute correspondence through attention map alignment. *ArXiv*, abs/2306.08877, 2023. URL https://api.semanticscholar.org/CorpusID:259164750.

Ho Youn Seo, Junseo Bang, Haechang Lee, Joohoon Lee, Byung Hyun Lee, and Se-Young Chun. Geometrical properties of text token embeddings for strong semantic binding in text-to-image generation. 2025. URL https://api.semanticscholar.org/CorpusID:277451749.

Zhilei Shu, Shangwen Zhu, Zihang Liang, Xiaofan Li, Qianyu Peng, Xinyu Cui, Bo Ye, Yiming Li, Fan Cheng, Jian Zhao, et al. Tie: Time interval encoding for video generation over events. *arXiv preprint arXiv:2605.10543*, 2026.

Kaiyue Sun, Kaiyi Huang, Xian Liu, Yue Wu, Zihan Xu, Zhenguo Li, and Xihui Liu. T2v-compbench: A comprehensive benchmark for compositional text-to-video generation. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 8406–8416. IEEE, 2025.

Ye Tian, Ling Yang, Haotian Yang, Yuan Gao, Yufan Deng, Jingmin Chen, Xintao Wang, Zhaochen Yu, Xin Tao, Pengfei Wan, et al. Videotetris: Towards compositional text-to-video generation. *Advances in Neural Information Processing Systems*, 37:29489–29513, 2024.

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

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. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 13629–13638, June 2025.

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 *CVPR*, 2025.

Qianxun Xu, Chenxi Song, Yujun Cai, and Chi Zhang. Switchcraft: Training-free multi-event video generation with attention controls. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 29136–29145, June 2026.

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

Xingyi Yang and Xinchao Wang. Compositional video generation as flow equalization. *ArXiv*, abs/2407.06182, 2024. URL https://api.semanticscholar.org/CorpusID:271050333.

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, Xu Bin, Xiaotao Gu, Yuxiao Dong, and Jie Tang. Cogvideox: Text-to-video diffusion models with an expert transformer. In Y. Yue, A. Garg, N. Peng, F. Sha, and R. Yu (eds.), *International Conference on Learning Representations*, volume 2025, pp. 83048–83077, 2025b. URL https://proceedings.iclr.cc/paper_files/paper/2025/file/ce31378e9f41d8907e97dab172b6c559-Paper-Conference.pdf.

Serena Yeung, Olga Russakovsky, Ning Jin, Mykhaylo Andriluka, Greg Mori, and Li Fei-Fei. Every moment counts: Dense detailed labeling of actions in complex videos. *International Journal of Computer Vision*, 126(2):375–389, 2018.

Hongyu Zhang, Yufan Deng, Zilin Pan, Peng-Tao Jiang, Bo Li, Qibin Hou, Zhen Dong, Zhiyang Dou, and Zhou Daquan. Ts-attn: Temporal-wise separable attention for multi-event video generation. In C. Vondrick, B. Hariharan, C. Raffel, L. Pinto, D. Yang, and A. Faust (eds.), *International Conference on Learning Representations*, volume 2026, pp. 61437–61462, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/file/6427302778de982f3bb70fa73dfc7861-Paper-Conference.pdf.

Dian Zheng, Ziqi Huang, Hongbo Liu, Kai Zou, Yinan He, Fan Zhang, Lu-Lu Gu, Yuanhan Zhang, Jingwen He, Wei-Shi Zheng, Yu Qiao, and Ziwei Liu. Vbench-2.0: Advancing video generation benchmark suite for intrinsic faithfulness. *ArXiv*, abs/2503.21755, 2025. URL https://api.semanticscholar.org/CorpusID:277350030.

A Benchmark and Generation Settings

Splits. Our benchmark consists of 84 prompts in two splits, which Table 5 summarizes together with the single-event prompts used in the analysis. The mixed split contains 54 prompts with 493 requested actions, up to three subjects per prompt, and up to six co-required actions per event, and each prompt comes with a first frame and a segmentation mask of every subject for image-to-video generation. The sequential split contains 30 prompts with consecutive events collected from the results of MinT (Wu et al., 2025). Under our grouping into events, 47% of its events contain a single action, against 30% in the mixed split, and instructions about the camera are also judged as actions. For the analysis we derive 20 single-event prompts with 47 actions from the mixed split (Appendix B).

Table 5: Statistics of the two splits of our benchmark and of the single-event prompts derived from the mixed split for the analysis. Events are formed per subject as maximal intervals over which the set of active actions does not change.

				Events by co-required actions		
Split	Prompts	Actions	Subjects per prompt	1	2	3 or more
Mixed	54	493	1 (17), 2 (34), 3 (3)	91	142	66
Sequential	30	265	1 (19), 2 (8), 3 (3)	70	52	27
Single-event prompts	20	47	1 (20)	–	13	7

Generation. The analysis uses Wan2.2 TI2V-5B in the image-to-video mode at 832×480 resolution with 81 frames at 16 frames per second and 40 denoising steps. The main results additionally use the text-to-video mode, Wan2.2 A14B with the same resolution, number of frames, and number of steps, and CogVideoX1.5-5B at 720×480 resolution with 81 frames at 16 frames per second and 50 denoising steps. The scale of classifier-free guidance is 4.0 for Wan2.2 TI2V-5B and for the first expert of Wan2.2 A14B, 3.0 for its second expert, and 6.0 for CogVideoX1.5-5B.

B Single-Event Prompts

For each of 20 events with a single subject and several co-required actions, we write a prompt that describes only this event over the entire clip and reuse the first frame and subject mask of the source prompt. For every prompt we also write one prompt per action that requests the action alone with the same description of the subject, which yields 47 single-action prompts. Thirteen prompts contain two actions and seven contain three. Seven of the events start after the first frame in their source prompt, so the first frame may show a posture that precedes the event, and we mark them in Table 6. Table 6 lists all prompts.

C Evaluation Protocol

Queries. Each action is judged within its own interval. Actions whose intervals coincide are judged together in one query, and actions whose intervals only partially overlap are judged in separate queries. Each query contains the first frame of the generated video, in which every designated subject is marked by an outline of a distinct color, followed by up to 24 frames sampled uniformly from the interval and the full prompt (Fig. 5). The judge does not see the rest of the video. The outline is drawn without filling the region, since a filled region changes the apparent clothing of the subject and leads the judge to treat it as another person. Where no subject mask is available, the designated subject is given by its description in the prompt. Each action is described by its verb and object, its designated subject, and its kind, namely a body movement, a facial expression or emotion, speaking, or a held state or posture, so that faint expressions and held states are judged with the appropriate criterion.

Responses. For every action the judge returns who performs it, whether it is visibly performed in the given frames, whether the performer is the designated subject, and the range of frames over which

Table 6: Single-event prompts. Each action is also requested alone with the same description of the subject and the same first frame. Prompts marked $*$ come from an event that starts after the first frame in the source prompt.

#	Prompt requested together	Actions
1	A woman in a black sequined dress and pearl necklace laughs and claps while standing up.	3
2	A woman in a black sequined dress and pearl necklace cries and covers her mouth while standing up.	3
3	A man in a blue shirt stands up from a table while drinking from a bottle with one hand and lifting a bread bag with the other hand.	3
4	A man walks around while touching his hair.	2
5	A woman in a light blue dress turns around while holding her dress with both hands.	2
6	A woman in a light blue dress walks down an aisle while smiling.*	2
7	A man sits down on the floor while hugging a cushion with one hand and lifting a pen with the other hand.	3
8	A man writes on a piece of paper while scratching his head.*	2
9	A person wearing a bandana and glasses, seated at a table, reads a book and turns a page while covering their mouth with one hand.	3
10	A person wearing a bandana and glasses stands up from the table while taking off their glasses.*	2
11	A person sits on a bed, swinging their legs while typing on a laptop.	2
12	A man seated at a table sets a cup down while closing a book.	2
13	A man wipes the table with a tissue with one hand while touching his hair with the other hand.*	2
14	A man holding a glass swirls it while sniffing it.*	2
15	A girl in a checkered dress looks up, holding a tree branch with one hand and picking an apple with the other hand.	3
16	A man in a top hat looks around, pointing forward with one hand and raising his other hand.	3
17	A woman places a telephone on a nightstand with one hand and turns off a lamp with the other hand.	2
18	A woman talks while lifting a piece of paper with one hand.*	2
19	A man sets the telephone down while taking a cigarette out of his mouth.	2
20	A man in a blue shirt is making a phone call while touching his beard.*	2

```
The FIRST image is frame 0 of the video. In it, the people that the actions must be
performed by are marked with a coloured outline: {legend}. The outline only marks WHO;
it does not change how they look. Keep track of each outlined person through these frames
by their position, clothing and appearance.

After that first image, these {n} images are ALL the frames of one time interval of
a generated 81-frame video (16 frames per second). They are, in order, video frames
{frame_list} --- that is, video frames {s} to {t}. You are NOT shown the rest of the video.

The video was generated from this prompt: "{prompt}"

The prompt requires the following actions to be performed during exactly this interval
(video frames {s} to {t}).
{items}

For every action give four fields, in this order.
- WHO: the person who actually performs this action in these frames. If it is one of the
outlined people, write the COLOUR of their outline (for example RED). If it is somebody
else, describe them in a few words by position and clothing. If nobody performs it, write
NONE.
- PRESENT: decide yourself whether the action is visibly performed somewhere inside these
frames. Write yes only if you can see it here. For a body movement, the movement itself
must be seen, not just a pose before or after it. A generated video may render a facial
expression, speech or a held state faintly; count it if it is recognisable, but do not
guess when the face or hands cannot be seen.
- SUBJECT: write yes if WHO is the required outlined person, no if a different person
performs it (including a person with a different outline colour), unclear if you cannot
tell. If PRESENT=no, write NA.
- FRAMES: the part of THIS interval that the action fills, as START-END video frame
numbers taken from the list above (both between {s} and {t}). Give the real extent inside
the interval --- the frames where the action is actually happening --- not the whole
interval by default. If PRESENT=no, write NONE.

Output exactly one line per action, in order, and nothing else:
ACTION <number>: WHO=<text> PRESENT=<yes/no> SUBJECT=<yes/no/unclear/NA> FRAMES=<START-END
or NONE>
```

Figure 5: Judge prompt. Fields in braces are filled per query. Each item names the action, its designated subject by outline color, and its kind (a body movement, a facial expression or emotion, speaking, or a held state or posture).

it is performed. We use Qwen3.8-27B served by vLLM, with temperature 0, seed 42, reasoning disabled, and at most 1536 output tokens. When a response cannot be parsed, we re-query up to three more times at temperature 0.5, and an action whose query remains unparsed counts as absent. No query of the $37{,}035$ judged actions remained unparsed.

Omission rate. For an action a with interval τ_a of length L_a frames, the judge returns whether a is performed, $p_a \in \{0, 1\}$, whether its performer is another subject, $o_a \in \{0, 1\}$, and a range of frames $\hat{\tau}_a \subseteq \tau_a$ over which it is performed. A performer reported as unclear is not counted as another subject, so $o_a = 0$ in that case. The coverage of a is $c_a = |\hat{\tau}_a|/L_a$, and a is realized when

$$r_a = p_a\,(1 - o_a)\,\mathbb{1}[c_a \geq \theta] = 1. \tag{8}$$

For N prompts, each with requested actions $\mathcal{A}^{(n)}$, the omission rate is

$$\mathrm{OM} = \frac{1}{N}\sum_{n=1}^{N}\frac{1}{|\mathcal{A}^{(n)}|}\sum_{a\in\mathcal{A}^{(n)}}\big(1 - r_a\big). \tag{9}$$

Every omitted action falls into exactly one of three categories. It is absent when $p_a = 0$, partial when $p_a = 1$, $o_a = 0$, and $c_a < \theta$, and performed by another subject when $p_a = 1$ and $o_a = 1$. The absent rate and the partial rate are computed with the same averaging over prompts as OM, and the three category rates sum to OM. In the main results, $\mathrm{OM_P}$ is the absent rate and $\mathrm{OM_S}$ adds the rate of actions performed by another subject.

Overlap criterion. Coverage is judged for each action separately, and the overlap among co-required actions follows from the threshold. If two actions share an interval of length L and each covers at least θL of it, their covered frames overlap for at least $(2\theta - 1)L$, which is 40% of the interval at $\theta = 0.7$. Two co-required actions performed largely one after another therefore cannot both be realized. For three actions sharing an interval, the overlap common to all of them is guaranteed only for $(3\theta - 2)L$, which is 10%. Table 7 reports the binding conditions of the analysis at $\theta = 0.5$. Interval masking is the only binding method that lowers the omission rate significantly at both thresholds, by 7.2 points at $\theta = 0.7$ and by 7.0 points at $\theta = 0.5$ (95% interval $[-12.7, -1.4]$). Table 8 repeats the pooled comparison of Sec. 4.2 at $\theta = 0.5$. Since $\mathrm{OM_P}$ and $\mathrm{OM_S}$ do not depend on coverage, only the omission rate changes with the threshold. At $\theta = 0.5$, the omission rate of CoRAct is significantly lower than that of every baseline.

Table 7: Omission rate (%) of the binding conditions on Wan2.2 TI2V-5B at $\theta = 0.7$, used throughout, and at $\theta = 0.5$.

	Mixed		Sequential	
Condition	$\theta = 0.7$	$\theta = 0.5$	$\theta = 0.7$	$\theta = 0.5$
No binding	68.9	59.2	38.7	35.6
TS-Attn	64.3	57.3	39.5	35.9
SwitchCraft	66.6	60.5	37.1	29.3
Interval masking	61.6	52.3	34.8	28.0

Table 8: Differences pooled over the six settings of the main results at $\theta = 0.5$, in points with bootstrap 95% intervals over prompts. Bold values have intervals that exclude zero. $\mathrm{OM_P}$ and $\mathrm{OM_S}$ equal those at $\theta = 0.7$ (Table 17).

Comparison	OM	$\mathrm{OM_P}$	$\mathrm{OM_S}$
CoRAct − Backbone	$\mathbf{-3.3}$ [−5.6, −1.1]	$\mathbf{-2.8}$ [−4.9, −0.6]	$\mathbf{-2.9}$ [−5.1, −0.5]
CoRAct − TS-Attn	$\mathbf{-2.5}$ [−4.4, −0.6]	$\mathbf{-2.2}$ [−4.3, −0.1]	−2.1 [−4.3, +0.1]
CoRAct − SwitchCraft	$\mathbf{-3.2}$ [−5.6, −0.7]	$\mathbf{-3.2}$ [−5.5, −0.8]	$\mathbf{-3.2}$ [−5.7, −0.7]
TS-Attn − Backbone	−0.8 [−3.0, +1.3]	−0.6 [−2.5, +1.4]	−0.8 [−3.0, +1.3]
SwitchCraft − Backbone	−0.1 [−2.6, +2.3]	+0.4 [−2.0, +2.8]	+0.3 [−2.2, +2.8]

Consistency of the judge. To measure how consistently the judge decides, we compare two independent runs of the judge on the same generated videos of three conditions of the mixed split, which gives 1341 pairs of decisions on the 5B model. Table 9 reports the agreement. The two runs agree on whether an action is realized at $\kappa = 0.82$ and on the four outcomes at $\kappa = 0.79$. Deciding whether an action appears is not more consistent than deciding whether it is realized, so we do not treat the appearance of an action as a more reliable measure than its coverage.

Table 9: Agreement between two runs of the judge on the same videos (no binding, TS-Attn, and SwitchCraft on the 5B model, 1341 actions). The four outcomes are realized, absent, partial, and performed by another subject.

Decision	Agreement	Cohen's κ
Appears in the interval	91.4%	0.81
Realized	92.8%	0.82
Four outcomes	86.1%	0.79

Order of events. We follow the two-stage template of StoryEval (Wang et al., 2025) with the same judge at temperature 0. Each video is judged twice, an event counts as completed only when both runs judge it completed, and the score is averaged over prompts. A run whose response cannot be parsed after three attempts yields no events, and a prompt without two valid runs is excluded from the average, which happened once in 15,888 runs.

Human validation. Nine annotators each judged two prompts, drawn from 10 prompts of our benchmark, and for every prompt watched the videos of the four methods on Wan2.2 TI2V-5B in random order under anonymous labels. For every event of the prompt, they marked whether it is performed as requested, and every prompt was judged by one to four annotators. On the events judged by several annotators, the annotators agree on 84.3% of their decisions, and the judge agrees with the annotators on 64.4% of the events. The judge is more lenient than the annotators, but it ranks CoRAct, SwitchCraft, and TS-Attn in the same order. Under human judgment, CoRAct performs 31.1% of the events as requested, against 19.6% for TS-Attn, 27.7% for SwitchCraft, and 5.4% for TIE. Paired over the same annotator and event, CoRAct exceeds TS-Attn by 11.5 points (McNemar $p = 2.3 \times 10^{-3}$) and TIE by 25.7 points ($p < 10^{-7}$), and its difference from SwitchCraft is not significant (+3.4 points, $p = 0.49$).

D Omission Rate and StoryEval

StoryEval (Wang et al., 2025) judges a video against a list of event sentences. A vision-language model first describes keyframes of the whole video and then decides, from the description, whether each event is completed. When our prompts are converted into this form, each event collects every action that is ongoing in its time span, so a single long action belongs to several events, and its omission fails all of them at once. The judge decides each event as a whole, which lets it accept an event with a missing action inferred from the scene and reject an event whose actions all appear because of a mismatched detail. Moreover, the protocol does not examine when an action occurs, so an action that appears outside its interval still counts. Our protocol instead judges each action within its own interval and by its designated subject, and therefore distinguishes the concurrent actions that are performed as requested from those that only appear.

E Binding and Encoding Conditions

Table 10 lists the binding conditions of the analysis, and Table 11 reports paired differences over prompts with bootstrap 95% intervals. On the mixed split, interval masking lowers the omission rate significantly and also reduces absent actions by a few points, with an interval that barely excludes zero. No binding method changes partial actions or the order of events significantly on the mixed split, and none changes any measure significantly on the sequential split.

Isolating events inside the encoder is implemented in two ways. In the strict version, the tokens of each event attend only to their own event and to the shared description, and the shared description attends only to itself, so the subject tokens see no action. In the relaxed version, the shared description attends to every token, so only the direct attention between actions of different events is blocked. On the sequential split, the strict version raises absent actions significantly, and the relaxed version raises them by 3.7 points fewer than the strict version (95% interval $[-7.3, -0.4]$). On the mixed split, the strict version raises absent actions from 31.6% to 35.5%. Neither version lowers the omission rate on either split.

Table 10: Binding conditions on Wan2.2 TI2V-5B. Rates are in %, and Other is the rate of actions performed by another subject.

Split	Condition	Omission	Absent	Partial	Other	Order
Mixed	No binding	68.9	36.8	31.1	1.0	0.178
	TS-Attn	64.3	35.7	28.1	0.5	0.195
	SwitchCraft	66.6	36.4	29.3	1.0	0.150
	Interval masking	61.6	31.6	29.6	0.5	0.170
	+ events isolated (strict)	68.8	35.5	32.1	1.2	0.077
	+ events isolated (relaxed)	66.0	30.0	35.0	1.0	0.119
Sequential	No binding	38.7	25.7	11.6	1.3	0.256
	TS-Attn	39.5	24.6	13.2	1.6	0.231
	SwitchCraft	37.1	22.4	13.7	1.0	0.292
	Interval masking	34.8	20.2	12.6	2.0	0.308
	+ events isolated (strict)	39.9	28.2	9.8	1.9	0.330
	+ events isolated (relaxed)	38.4	24.4	11.9	2.0	0.258

Table 11: Paired differences over prompts with bootstrap 95% intervals on Wan2.2 TI2V-5B. Rates are in points and order is on its native scale. Rows marked † are relative to interval masking, and the others relative to no binding. Bold values have intervals that exclude zero.

Split	Condition	Omission	Absent	Partial	Order
Mixed	TS-Attn	$-4.6\,[-9.6, +0.7]$	$-1.1\,[-6.2, +4.1]$	$-3.0\,[-8.4, +2.6]$	$+0.017\,[-0.054, +0.091]$
	SwitchCraft	$-2.3\,[-7.2, +2.7]$	$-0.4\,[-6.5, +5.8]$	$-1.8\,[-7.2, +3.9]$	$-0.028\,[-0.102, +0.050]$
	Interval masking	$\mathbf{-7.2}\,[-12.5, -2.0]$	$\mathbf{-5.2}\,[-10.3, -0.1]$	$-1.5\,[-6.5, +3.3]$	$-0.008\,[-0.077, +0.064]$
Sequential	TS-Attn	$+0.9\,[-3.1, +4.9]$	$-1.1\,[-5.5, +3.3]$	$+1.6\,[-1.7, +5.0]$	$-0.017\,[-0.115, +0.086]$
	SwitchCraft	$-1.6\,[-8.9, +5.9]$	$-3.3\,[-10.0, +3.2]$	$+2.1\,[-3.4, +7.5]$	$+0.046\,[-0.032, +0.141]$
	Interval masking	$-3.8\,[-9.7, +2.1]$	$-5.5\,[-11.1, +0.2]$	$+1.0\,[-2.9, +5.2]$	$+0.039\,[-0.039, +0.124]$
	Events isolated (strict)†	$+5.0\,[-0.5, +10.7]$	$\mathbf{+8.0}\,[+2.9, +13.5]$	$-2.9\,[-7.7, +2.1]$	$+0.022\,[-0.086, +0.136]$
	Events isolated (relaxed)†	$+3.5\,[-0.8, +7.6]$	$+4.2\,[-0.0, +8.5]$	$-0.7\,[-6.6, +5.0]$	$-0.050\,[-0.139, +0.042]$

Replication on Wan2.2 I2V-A14B. Table 12 repeats the comparison of temporal binding on the mixed split with the A14B model, from the run of the main results. As on the 5B model, no binding method changes the absent or the partial rate significantly, and interval masking lowers the omission rate the most.

Table 12: Temporal binding on Wan2.2 I2V-A14B, mixed split, over the 50 prompts judged with the same labels under every condition. Levels in % and differences relative to no binding in points with bootstrap 95% intervals.

Condition	Omission	Absent	Partial	Δ Absent	Δ Partial
No binding	64.1	12.4	51.8		
TS-Attn	64.2	14.3	49.7	$+2.0\,[-2.9, +7.3]$	$-2.1\,[-8.8, +5.1]$
SwitchCraft	60.5	11.7	48.7	$-0.7\,[-4.6, +3.2]$	$-3.1\,[-9.3, +3.1]$
Interval masking	58.2	10.1	48.0	$-2.3\,[-6.4, +1.7]$	$-3.8\,[-10.2, +2.2]$

F Encoder Analysis

Representation shift. For every action of a prompt we compare its token representations when the full prompt is encoded jointly and when its event is encoded alone, and measure the cosine distance at the final layer averaged over its tokens. To vary the number of accompanying events, we encode the target event together with 0 to 5 other events of the same prompt. On the mixed split, the distance grows monotonically with the number of accompanying events in both encoders, from 0.022 to 0.041 for umT5 and from 0.080 to 0.135 for T5 as the accompanying events increase from one to five. On the sequential split, it rises from 0.061 to 0.093 for umT5 and from 0.184 to 0.307 for T5 over one to three accompanying events and then levels off.

Event-specific shift per action. For each of the 493 actions of the mixed split we compare three encodings, in which the actions of the other events are present, hidden inside the encoder, or replaced by the same number of neutral words. We measure the cosine distance of the final representation of the action tokens from the hidden encoding. The distance is larger with the other events present than with neutral words for 88.2% of the actions in umT5 (medians 0.233 and 0.127, Wilcoxon $p = 3.8 \times 10^{-56}$) and for 71.0% in T5 (medians 0.187 and 0.138, $p = 8.8 \times 10^{-22}$).

Mentions of a subject. For each of the 290 events of the mixed split whose subject has other events, we average the self-attention that a mention of the subject gives to the verb tokens of its current event, of the same subject's other events, and of other subjects under joint encoding. We read it from the first mention of the subject and from the mention inside the event. Table 13 reports both encoders. The first mention attends to the verbs of the subject's other events as much as to those of its current event, and much less to the verbs of other subjects, whereas the mention inside an event favors its own event. Interval masking hides neither mention from any frame, since every frame must read the subject to know who acts (Appendix A).

Table 13: Self-attention from a subject mention to verbs under joint encoding, averaged over all layers. Ratio is the attention to the verbs of the subject's other events over that to the verbs of its current event.

Encoder	Mention	Current event	Other events	Other subjects	Ratio	Current event favored
umT5	First	0.0266	0.0268	0.0178	1.008	44.5%
	Inside the event	–	–	–	0.678	75.5%
T5	First	0.0234	0.0233	0.0126	0.997	43.8%
	Inside the event	–	–	–	0.584	76.9%

G Attention Diagnostics

Measurement. The original single-event videos were not available together with their attention, so we generated them again with the settings of Appendix A and judged them again with the protocol of Appendix C. The prompts are the 20 single-event prompts, in which each action is requested together with the others, and 47 prompts that request each of their actions alone. For every latent frame we record the cross-attention of the conditional branch, averaged over the first 10 denoising steps, over blocks 10 to 19, and over heads. The subject map is the attention to the row of the subject that the attention-side component uses, normalized by its maximum within the frame, and the support $q_{a,f}$ follows its definition in Sec. 2.4. An action is performed at a latent frame when the judge reports it as performed by its subject over frames that overlap that latent frame. The correlation between the share of support and the performed action is computed over the latent frames of each pair of co-required actions, and pairs in which the share or the difference of the performed indicators does not vary are excluded.

Individual and concurrent requests. On the 5B model, 44.7% of the actions are omitted when requested together and 48.9% when requested alone, and the difference is not significant (-4.3 points, 95% interval $[-20.8, +11.4]$, exact McNemar $p = 0.77$). The actions in these prompts are thus often omitted even when requested alone. Their support, however, depends on the request. The same action receives less support when it is requested together with others, retaining 0.78 of its support on average (Table 14).

Support of co-required actions. Table 14 summarizes the relation between the support and the performed actions. An action receives less support when requested together than alone, receives more support in the frames where it is performed, and the share of support held by each of two co-required actions tracks which of them is performed.

Raising an omitted action. This experiment uses the original single-event videos and their judgments. We add $\log \alpha$ to the logits of the tokens of one omitted action, the one with the lowest coverage under the concurrent request, in the conditional branch during the first 10 denoising steps, either at every location or within the subject region, which consists of the locations where the subject

Table 14: Support of co-required actions on the single-event prompts (5B), from regenerated videos judged again.

Analysis	Result	Wilcoxon p
Support, together over alone (47 actions)	0.78 [0.68, 0.91], below one for 35	6.7×10^{-4}
Support in performed vs. other frames (21 actions)	.0068 vs. .0060, higher for 16	1.4×10^{-3}
Share of support vs. performed action (15 pairs)	mean $r = 0.30$, positive for 12	8.4×10^{-3}

map exceeds half of its maximum over the frame. On three prompts we compared $\alpha \in \{4, 8\}$ and blocks 10 to 19 or 0 to 14, and we use $\alpha = 4$ with blocks 10 to 19. Table 15 reports the results on the 12 prompts whose concurrent request failed. Raising the support lowers the omission rate of these prompts, but most raised actions still cover less than θ of their interval, since a single raise favors one action while the competition recurs at every frame. Of the 11 co-required actions that were realized before raising, 2 fail after raising at every location and 1 after raising within the subject region.

Table 15: Raising the attention of an omitted action on 12 failed single-event prompts (5B). The omission rate is over all actions of these prompts, and the counts refer to the raised action.

Scaling	Omission (%)	Appears	Realized	Partial
None	65.3	–	–	–
Every location	52.8	12/12	4/12	8/12
Subject region	56.9	11/12	3/12	8/11

H Components of Omission Across Methods

We separate the actions of the mixed split into single actions, which overlap with no other action of the same subject, and co-required actions. Table 16 reports both groups for the methods of the main results, pooled over the six settings of Sec. 4.1. About half of the appearing actions are not sustained over their interval, nearly equally for single and co-required actions and under every method, including ours. Single actions go missing more often than co-required ones, which suggests that they include harder actions such as brief posture transitions, so we do not compare the two groups to attribute omission to concurrency.

Table 16: Absent rate, partial rate, and the fraction of appearing actions that are not sustained, for single and co-required actions, pooled over the six settings of the main results (%).

	Single			Co-required		
Method	Absent	Partial	Not sustained	Absent	Partial	Not sustained
Backbone	48.0	28.1	54.8	34.7	35.7	55.4
TS-Attn	46.1	28.3	53.0	33.4	36.1	54.8
SwitchCraft	42.6	28.8	50.9	34.9	35.8	55.7
CoRAct	40.8	31.4	53.4	31.4	37.4	54.9

Table 17 reports all differences pooled over the six settings, including the omission rate, which is significantly lower for CoRAct than for the backbone and does not differ significantly from those of TS-Attn and SwitchCraft.

On the 5B model in the image-to-video mode, the attention-side component accounts for the reduction of absent actions. Relative to the text-side component with interval masking, adding it lowers absent actions by 4.7 points (95% interval $[-8.8, -0.6]$) and raises partial actions by 5.2 points ($[+0.7, +9.7]$), leaving the omission rate unchanged ($+0.2$, $[-5.4, +5.5]$).

Table 17: All differences pooled over the six settings of Table 2, in points with bootstrap 95% intervals over prompts. Bold values have intervals that exclude zero.

Comparison	OM	OMP	OMS
CoRAct − Backbone	**−2.5** [−4.8, −0.3]	**−2.8** [−4.9, −0.6]	**−2.9** [−5.1, −0.5]
CoRAct − TS-Attn	−1.5 [−3.4, +0.3]	**−2.2** [−4.3, −0.1]	−2.1 [−4.3, +0.1]
CoRAct − SwitchCraft	−1.7 [−3.9, +0.7]	**−3.2** [−5.5, −0.8]	**−3.2** [−5.7, −0.7]
TS-Attn − Backbone	−1.0 [−3.1, +1.1]	−0.6 [−2.5, +1.4]	−0.8 [−3.0, +1.3]
SwitchCraft − Backbone	−0.9 [−3.0, +1.2]	+0.4 [−2.0, +2.8]	+0.3 [−2.2, +2.8]

I Implementation Details

Text-side component. We encode the prompt once with the text encoder of the backbone and create a copy of the output for every segment σ_u. Within each copy, the representations of different subjects are first orthogonalized with symmetric Löwdin orthogonalization over the rows of each subject, which preserves the norm of every row and the inner products within a subject. For every action a of an event e_k active in the segment, we place the sum of the input embeddings of its verb and object tokens at the input row p_k of its subject and compute the response $c_{k,a}$ of the output row with a Jacobian-vector product at the original input. When the subject of the action is a pronoun, the pronoun row is used instead of the head noun. The response is projected onto the complement of the subject direction $\hat{h}_k$ and bounded as in Eq. (equation 2) with $\rho = 0.1$. The directions $b_{k,a}$ of the co-required actions are averaged into m_k, and the subject row is rotated toward m_k by the angle $\phi_k = \phi_{\max}\|m_k\|$ with $\phi_{\max} = 45°$ while its norm is preserved. The verb phrases of actions that are inactive in the segment are hidden from its frames.

Attention-side component. Guidance is applied at the end of each of five early denoising steps, which form the set $\mathcal{S}$. We run the conditional branch up to the last guided block with the latent requiring gradients and collect, for every segment, the attention of its latent frames to the verb tokens of each active action and to the rows of each subject in blocks 10 to 19 for Wan2.2 TI2V-5B and 13 to 21 for Wan2.2 A14B. The maps are averaged over tokens, heads, and blocks and smoothed with a 3×3 Gaussian. The subject maps are detached and normalized by their maximum at each frame. The temperature T_f of the soft minimum is 0.1 times the mean support at frame f, and we set $\lambda_p = 0.1$, $\lambda_e = 0.5$, and $\epsilon = 10^{-6}$. The gradient of $\mathcal{L}_u$ is applied only to the latent frames of σ_u through M_u, the first latent frame is kept fixed in the image-to-video mode, and the gradient is smoothed over time within each segment with the kernel $[1, 2, 1]/4$. The step size η_t starts at 20 and decays linearly to 4 over the five steps, and the update of each frame is clipped to 0.005 times its norm with the same decay.

CogVideoX. The text-side component is identical. Since CogVideoX processes text and video tokens jointly, we replicate the text stream once per segment, and the video tokens of each segment attend to the image tokens and to the unhidden rows of their own stream. The attention maps for guidance are obtained by renormalizing the attention from video to text tokens over the text keys, guidance is applied in blocks 14 to 27 of 42, and the gradient of each temporal token is distributed to the two latent frames it covers.

J Computational Cost

We measure the time to generate one video and the peak GPU memory in image-to-video generation on a single NVIDIA H100 GPU with the settings of Appendix A. Table 18 reports the results. CoRAct adds 15 seconds per video to Wan2.2 TI2V-5B and takes about 50% longer than Wan2.2 A14B, and on both models it roughly doubles the peak memory, since the guided steps backpropagate through the guided blocks. On CogVideoX1.5-5B, it keeps the peak memory of the backbone but takes about three times as long. SwitchCraft remains slower than CoRAct on Wan2.2 TI2V-5B and on CogVideoX1.5-5B.

Table 18: Time to generate one video and peak GPU memory in image-to-video generation on a single H100 GPU. *TIE is trained with event timestamps, and its training cost is not included.

Backbone	Method	Time (s)	Memory (GB)
CogVideoX1.5-5B	Backbone	90.5	40.9
	TS-Attn	111.5	40.9
	SwitchCraft	342.9	39.5
	CoRAct	269.0	40.9
Wan2.2 TI2V-5B	Backbone	67	17.0
	TS-Attn	68	17.0
	SwitchCraft	409	17.0
	TIE	70*	29.2
	CoRAct	82	37.4
Wan2.2 A14B	Backbone	494	34.5
	TS-Attn	515	38.5
	CoRAct	733	66.9

K LIMITATIONS

Our method reduces the actions that never appear in their interval, but about half of the actions that appear are still not sustained over it, as under every other method we evaluate. This may reflect the temporal behavior of current backbones, intervals that are longer than the natural duration of some actions, or the way the judge delimits an action in time, and separating these factors is left to future work. Our analysis covers encoders of the T5 family, and encoders based on multimodal language models are left to future work. Interval masking hides only the verb phrases of inactive actions, and the text-side component edits the first mention of each subject and the pronouns that refer to it, so a subject named again by a repeated noun phrase keeps its original representation in every interval. MM-DiT backbones such as CogVideoX re-contextualize text tokens with video tokens in every block, which dilutes an edit made to the encoder output.