DO VIDEO GENERATORS TRACK THE WORLD ACROSS SEGMENTS? A BENCHMARK AND METHOD FOR WORLD-STATE REASONING IN VIDEO CONTINUATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Video generators build long videos by composing shorter parts, either by generating segments one after another or by autoregressively extending chunks. Each new part usually depends on memories of historical observations, such as recent frames, selected key frames, memory banks, or cached features. These memories preserve visible evidence from the past, but current generators do not reliably turn such evidence into a world-state interface: what holds in the video world after previous actions and how it should change under the next prompt. A past frame remains valid history, but it may not describe the state needed by the next segment; some states must instead be inferred from occluded or implicit changes rather than copied from a directly observed frame. This creates a simple but overlooked question for video continuation: given a previous video, its prompt, and a new prompt, can a model generate a continuation that reflects the state determined by both the historical video and the new prompt? To answer this question, we introduce STATEBENCH, a benchmark that targets this gap by testing continuations over three state categories: past-visible states, occluded-process states, and complex-transition states. We further propose STATEAGENT, which explicitly maintains an entity-state representation, updates it under the new prompt, grounds the predicted post-action state as a future end frame, and renders the next video. Experiments show that our method improves controlled video continuation by raising the all-case state score (SCS-All) from 45.4 to 69.3, and also benefits story generation at the one-minute scale.

1 INTRODUCTION

As video generation aims for longer outputs and coherent visual stories, current systems face two practical choices. One is to generate a very long video in a generation process, which is computationally expensive and remains difficult for current models. The other is to generate shorter segments or chunks sequentially, where each new unit must be conditioned on the generated history. Providing the complete history to every new unit is also costly and often impractical, so existing continuation systems usually compress or select the history through recent frames, selected key frames (Zhou et al., 2024; Zhang et al., 2025), visual memory banks (Zhou et al., 2026; Xiao et al., 2026; Li et al., 2025; Wu et al., 2026b), or cached features (Chen et al., 2026b; Meng et al., 2026). These signals form an observation memory of the generated history, useful for preserving appearance, identity, etc.

Observation memory may contain evidence for the world state, but current continuation methods expose it as frames or features rather than as an updated state interface. Historical frames record what was visible at particular moments, while the next segment needs what should hold after the previous video and how it should change under the new prompt. A past frame is valid history, but not always current-state evidence: the relevant state may be hidden at the boundary, changed after an earlier observation, or inferred from an occluded process. We view this evidence-to-state conversion as a capability future video world models should internalize; under current model capabilities, we make it explicit through state reasoning.

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.

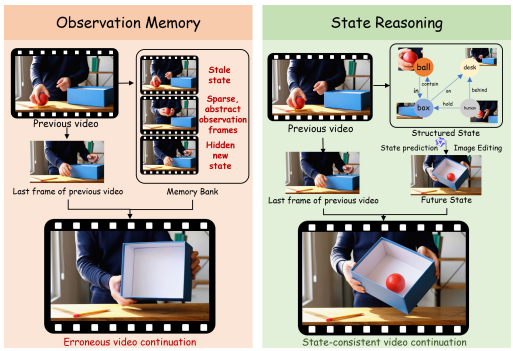

Figure 1: **Cross-segment state reasoning.** Observation memory stores visible evidence from previous frames, while state reasoning updates what should now be true after the historical action. After an object is placed into a container and becomes invisible, the next segment should know that the object is still inside.

Several recent works have started to study hidden state, out-of-sight content, or long-horizon world dynamics, but they address different settings from video continuation. StEvo-Bench studies hidden state evolution within a single generated video, but it requires the model to establish, hide, evolve, and reveal the state in one generation, making failures hard to attribute (Ma et al., 2026). PAN uses LLM latent features as world states for long-horizon dynamics. This is useful for abstract simulation, but its state lacks explicit appearance binding, making it difficult to preserve the visual identity of entities in the next generated segment (Xiang et al., 2025). LiveWorld stores persistent 4D spatiotemporal memory for out-of-sight content, which can preserve rich scene evidence but is costly and difficult to scale. Crucially for our setting, it still stores past observations in a richer space, rather than explicitly converting historical events into an updated state for the next video segment (Duan et al., 2026). These differences leave open how to evaluate and support state update in video continuation.

We study the basic unit of state-aware video continuation. Given a previous video V_t, its source prompt p_t, and a continuation prompt p_{t+1}, a model should infer the world state at time $t+1$ by updating the historical state according to the continuation prompt, rather than merely maintaining visual continuity with previous frames. We introduce StateBench to operationalize this setting. StateBench contains 200 tasks across three state categories: past-visible states, occluded-process states, and complex-transition states. For each task, we provide a task-specific checklist. Evaluation first checks whether the generated continuation completes the state-establishing action requested by the prompt, and then checks whether the resulting state is correct.

We further propose StateAgent as a practical solution under current model capabilities. Our key insight is that state-consistent continuation should decouple state reasoning from video rendering: an explicit state-reasoning role should decide the next state, while the video generator should render that predicted state into pixels. StateAgent implements this idea by building an entity-state representation from the previous video, updating it under the continuation prompt to obtain the post-action state, grounding that state as a future end frame using an image editor, and rendering the continuation. Experiments show that stronger observation memory does not make current generators reliably capture the required world state. Recent-frame, key-frame, memory-bank, and autoregressive baselines often preserve visible appearance, but still fail when the current state is hidden, changed after an earlier observation or produced by an unseen change. Overall, our contributions are:

- We identify the gap between observation memory and state reasoning in video continuation and introduce StateBench, a unit benchmark that tests whether generated continuations follow the state implied by the previous video and the new prompt.
- We propose StateAgent, which decouples state updating from video rendering by leveraging a VLM to parse historical states and predict future states, grounding them into future frames via an image editor, and rendering the video continuation through keyframe-to-video generation.
- We show that our method outperforms observation-based memory methods, not only improving controlled video continuation by raising the StateBench score from 45.2 to 69.3, but also benefiting long story generation.

2 Problem Formulation

2.1 Cross-Segment State Handoff

Let V_t be a previous video, p_t its source prompt, and p_{t+1} the prompt for the next segment. Current continuation interfaces usually pass history forward through an observation-memory interface $O_t =$

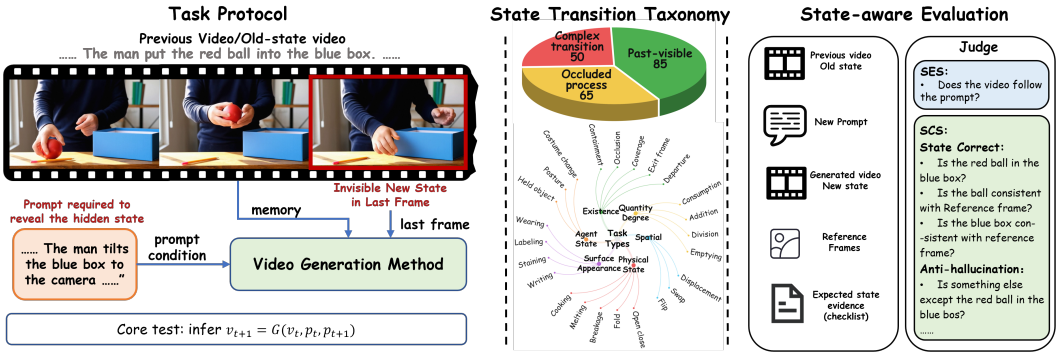

Figure 2: **StateBench overview.** Given a previous video V_t, its source prompt p_t, and a continuation prompt p_{t+1}, a method must generate V_{t+1} that reveals or uses the state established in the history. **Left: task protocol.** The previous segment contains the state-establishing event, while its tail frame, i.e., the generation boundary between two segments, hides or ambiguates the task-relevant relation. **Middle: state-evidence categories.** StateBench evaluates continuations across different ways in which the target state is evidenced by the history. **Right: State-aware evaluation.** SES first checks whether the requested reveal/revisit action is completed, and SCS then measures whether the exposed state is consistent with the previous video.

$M_{\text{obs}}(V_t)$, such as the last frame, recent frames, retrieved key frames, memory-bank entries, or cached temporal features. Here M_{obs} denotes the history interface actually exposed to the next generator, rather than the full information content of V_t. This interface exposes visual evidence in sparse, compressed, or potentially stale forms, making it difficult for current generators to infer a state that can be queried, updated, and grounded for rendering. The next segment is then generated as:

$$V_{t+1} \sim G(O_t,\ p_{t+1}). \tag{1}$$

The problem is therefore an interface gap. At the segment boundary, the observation exposed to the next generator may not make the post-history state explicit. In our setting, this boundary is the tail frame of the previous video: the historical event has established a task-relevant state, but the tail frame hides or ambiguates the relation. A state-aware continuation process should therefore preserve both visual evidence and an explicit post-history state belief:

$$S_t = E(V_t, p_t), \qquad V_{t+1} \sim G(O_t,\ S_t,\ p_{t+1}), \tag{2}$$

where O_t preserves visual evidence such as appearance, identity, layout, and style, while S_t denotes what should be physically true after the previous video. The central cross-segment state handoff problem occurs when the observation memory interface maps different histories (a) and (b) to similar exposed evidence, while their post-history states differ:

$$M_{\text{obs}}(V_t^{(a)}) \approx M_{\text{obs}}(V_t^{(b)}), \qquad S_t^{(a)} \neq S_t^{(b)}. \tag{3}$$

A generator that relies only on the exposed observation memory interface can struggle to distinguish these cases under current model capabilities, while an explicit state representation can condition the next segment on what is currently true.

2.2 State as Causal Belief

We define a world state as a set of task-relevant physical relations that may persist beyond visibility. The previous video and its prompt establish a post-history state S_t. The continuation prompt then induces a state update that may reveal, use, or further transform this state. We write this state reasoning step as:

$$S_{t+1} = \tau(S_t,\ a_{t+1}), \tag{4}$$

where a_{t+1} is the physical event described or implied by p_{t+1}. Here, S_{t+1} denotes the state that should be reflected in the continuation, and τ abstracts how the continuation prompt uses or updates

the state carried from the previous segment. Cross-segment generation therefore requires inferring S_t from the historical event and generating V_{t+1} according to the state implied by both S_t and p_{t+1}.

In STATEBENCH, we instantiate S_t as a set of task-relevant physical relations that expose the gap between observation and state:

$$S_t = \{r_i\}, \qquad r_i \in \mathcal{F}, \tag{5}$$

where $\mathcal{F}$ denotes the state types evaluated in STATEBENCH, including visibility, location, containment, transformation, surface and material state, quantity, and agent-object relations. These variables provide controlled tests of whether a continuation preserves the physical state implied by the previous segment.

2.3 OBSERVATION MEMORY VERSUS STATE REASONING

Observation memory asks whether generated content can reuse or resemble previous visible evidence; state reasoning asks whether current generators can infer and maintain the physical consequences of previous actions for the next segment. As illustrated in Figure 1, appearance continuity and state continuity are therefore complementary but distinct. A visual memory mechanism can preserve how entities looked, where they appeared, and how the scene was framed, but it is not a reliable state interface for deciding what should now be true after an action. This distinction motivates STATEBENCH, where continuations are evaluated not only by whether they follow the next prompt, but also by whether the revealed state is consistent with the previous video.

3 STATEBENCH: MEASURING STATE DISCONTINUITY

STATEBENCH is a compact unit test for state-aware video continuation. Each task provides a 5–10s previous video V_t, its source prompt p_t, and a continuation prompt p_{t+1}. The previous segment establishes a physical state, but its tail frame hides or ambiguates the task-relevant relation. The continuation must therefore use the state established before the segment boundary, rather than only the terminal appearance, to complete the requested state update.

3.1 TASK AND DATASET

STATEBENCH contains 200 cross-segment continuation cases across three state-evidence categories: 85 past-visible cases, 65 occluded-process cases, and 50 complex-transition cases. In every case, the task-relevant state is hidden or ambiguous at the generation boundary, and the next prompt induces an action that exposes, uses, or transforms this state without naming the target answer. Past-visible cases contain a frame that directly shows the target state before it becomes hidden. Occluded-process cases involve state changes inside a container, behind an obstruction, or otherwise out of direct view. Complex-transition cases involve harder physical or relational state transitions, such as spatial transfer, quantity change, color mixing, dissolution, or material and surface transformations. Figure 2 summarizes the task protocol, category split, and evaluation design. Detailed schemas, fine-grained categories, and prompt construction rules are provided in the appendix.

3.2 EVALUATION

Each generation method receives the same triplet (V_t, p_t, p_{t+1}) and generates one continuation $\hat{V}_{t+1}$. The evaluator receives $(V_t, p_t, p_{t+1}, \hat{V}_{t+1})$, a task-specific checklist, and reference frames selected from V_t for appearance-sensitive entities. These reference frames are used only for evaluation, not as additional generation inputs.

We use a two-stage checklist evaluation. *State Exposure Score* (SES) first measures whether the continuation completes the requested state-checking or state-updating action. For SES-positive samples, the state checklist contains state-correctness questions and anti-hallucination questions. Let $e_i \in \{0, 1\}$ indicate successful exposure, and let $y_{iq} \in \{0, 1\}$ denote whether the judge answers "yes" to checklist question q for sample i:

$$\text{SES} = \frac{1}{N} \sum_{i=1}^{N} e_i. \tag{6}$$

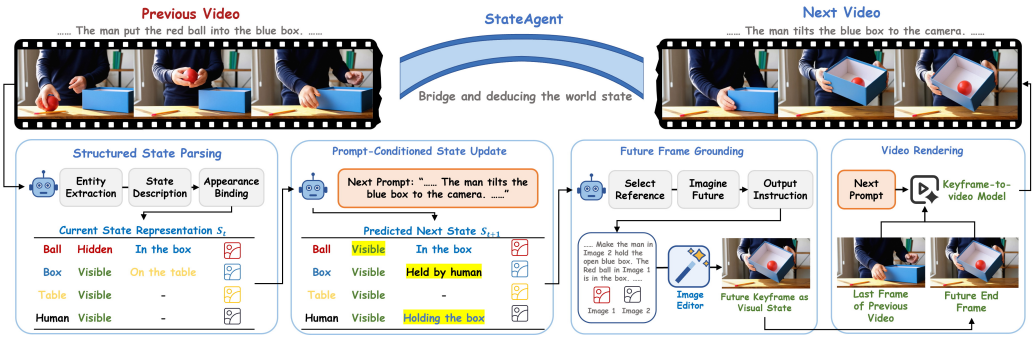

Figure 3: **StateAgent pipeline.** StateAgent decouples state prediction from video rendering. Given V_t, p_t, and p_{t+1}, it parses the historical event into a structured state graph, predicts the target post-action state, grounds it into a future end frame through image editing, and renders the continuation with first–last-frame video generation.

For SES-positive samples, *State Correctness* (SC) is the yes rate over state-correctness questions, while *Hallucination Rate* (HR) is one minus the yes rate over anti-hallucination questions:

$$\mathrm{SC} = \frac{\sum_i e_i \sum_{q \in \mathcal{C}_i} y_{iq}}{\sum_i e_i |\mathcal{C}_i|}, \qquad \mathrm{HR} = 1 - \frac{\sum_i e_i \sum_{q \in \mathcal{H}_i} y_{iq}}{\sum_i e_i |\mathcal{H}_i|}. \tag{7}$$

The conditional state consistency score is the yes rate over all state-checklist questions on SES-positive samples:

$$\mathrm{SCS}_{\mathrm{cond}} = \frac{\sum_i e_i \sum_{q \in \mathcal{C}_i \cup \mathcal{H}_i} y_{iq}}{\sum_i e_i |\mathcal{C}_i \cup \mathcal{H}_i|}. \tag{8}$$

We also report an all-case score, SCS-All, which treats exposure failure as an end-to-end state failure:

$$\mathrm{SCS}_{\mathrm{All}} = \frac{1}{N} \sum_i e_i c_i, \qquad c_i = \frac{1}{|\mathcal{C}_i \cup \mathcal{H}_i|} \sum_{q \in \mathcal{C}_i \cup \mathcal{H}_i} y_{iq}. \tag{9}$$

In the tables, SCS denotes $\mathrm{SCS}_{\mathrm{cond}}$ for readability.

Following criterion-specific VLM evaluation protocols used in other benchmarks (Bansal et al., 2024; Meng et al., 2024; Ma et al., 2026), scores are produced by a checklist judge. The judge is given the previous video, prompts, generated continuation, task checklist, and entity reference frames when appearance matching is required.

4 StateAgent: Future-Frame State Carrier

StateBench defines a capability that future video generation models should eventually internalize: carrying a world state across generation boundaries and updating it under a new prompt. Current video generators, however, do not reliably expose such a state interface. We therefore present StateAgent as a practical solution for the current stage of model capability. The method decouples *state prediction* from *state rendering*: an explicit predictor first infers what should be true after the continuation prompt, and a video generator then renders a segment constrained by that predicted state. StateAgent instantiates this state handoff formulation by making the intermediate state update explicit as shown in Figure 3. The parsed state graph represents S_t, the state after the previous segment. Given the continuation prompt, StateAgent predicts a target continuation state $\hat{S}_{t+1}$, grounds it as a future end frame, and then renders the next segment toward that predicted state.

This decoupling raises a representation question. Purely textual states are easy to update but too abstract to preserve appearance bindings; implicit latent states may support dynamics, but are difficult to inspect, costly to train, and can discard visual details needed for faithful continuation. StateAgent instead uses a *future end frame* as a visualized state carrier. A multimodal agent predicts the structured post-action state, an image-editing model grounds that state into pixels with historical appearance references, and a keyframe-to-video generator renders the motion between the

historical end frame and the predicted future end frame. This follows a similar intuition to ImageWAM, which uses image editing as a compact target-state transformation prior for world-action modeling (Zhang et al., 2026).

4.1 STRUCTURED STATE CARRIER

Given (V_t, p_t), STATEAGENT builds a lightweight state graph S_t rather than retrieved frames. Each entity node stores an identity, references to frames where the entity is visible, and a short state description that summarizes its current condition, such as whether it is open, broken, hidden, held, or stained. Relations record visibility, containment, location and so on. The graph is deliberately sparse: it stores only the state variables needed for future continuation, while leaving texture, pose, and motion synthesis to the renderer. Unlike key-frame memory, it can mark a relation as true even when not visible at the boundary, and it can mark an earlier visual observation as stale after a later action changes the state. When constructing S_t, STATEAGENT uses p_t only for entity binding between names and visual entities, while state parsing is grounded in the observed video evidence.

4.2 PROMPT-CONDITIONED STATE UPDATE

The continuation prompt is treated as an action condition that induces a state update. Given the parsed post-history state S_t and p_{t+1}, a state updater $\mathcal{U}$ predicts the target continuation state:

$$\hat{S}_{t+1} = \mathcal{U}(S_t,\ p_{t+1}). \tag{10}$$

Here, $\hat{S}_{t+1}$ is the method's explicit estimate of the state that should be reflected in V_{t+1}. It may reveal a hidden relation, use an existing relation, or further transform the physical state. The output specifies the target state for the continuation. This directly addresses the interface bottleneck of observation memory: hidden objects can remain in S_t, while stale appearances from earlier frames are not treated as the current state after later actions.

4.3 FUTURE-FRAME GROUNDING

A structured state alone is not enough to condition a video generator, because it lacks concrete appearance binding. STATEAGENT therefore grounds $\hat{S}_{t+1}$ into a future end frame $\hat{F}^{\text{end}}_{t+1}$. This step converts the symbolic state prediction into a visual condition that current video generators can follow. The historical end frame F^{end}_t provides camera, layout, and visible entities; reference frames R_t provide appearances for entities that are not visible at the boundary but are needed in the predicted state. A frame-grounding module $\mathcal{G}_{\text{frame}}$ receives the historical end frame, selected references, the predicted state, and the continuation prompt:

$$\hat{F}^{\text{end}}_{t+1} = \mathcal{G}_{\text{frame}}(F^{\text{end}}_t, R_t, \hat{S}_{t+1}, p_{t+1}). \tag{11}$$

The editing instruction is generated from the predicted state change and the selected visual references, so that the edited frame reflects the target state while preserving appearance and layout. The target frame is thus not a memory frame; it is a visualized prediction of the future state.

4.4 VIDEO RENDERING AND DIAGNOSTICS

Finally, a keyframe-to-video generator $\mathcal{G}_{\text{video}}$ renders the continuation:

$$\hat{V}_{t+1} = \mathcal{G}_{\text{video}}(F^{\text{end}}_t, \hat{F}^{\text{end}}_{t+1}, p_{t+1}), \tag{12}$$

where we instantiate $\mathcal{G}_{\text{video}}$ with the keyframe-to-video interface of Wan (Wan et al., 2025). The historical end frame keeps temporal continuity with the previous segment, while the future end frame constrains the state that must be reflected by the continuation. The renderer is therefore responsible for motion and visual realism, rather than rediscovering hidden physical state from scratch.

The modular structure makes failures inspectable. A wrong continuation can be attributed to history parsing, state update, future-frame grounding, or video rendering. This is useful because black-box video generation failures often look identical in the final video, while STATEAGENT exposes whether the system failed to carry the state, update it, bind it to appearance, or animate toward it.

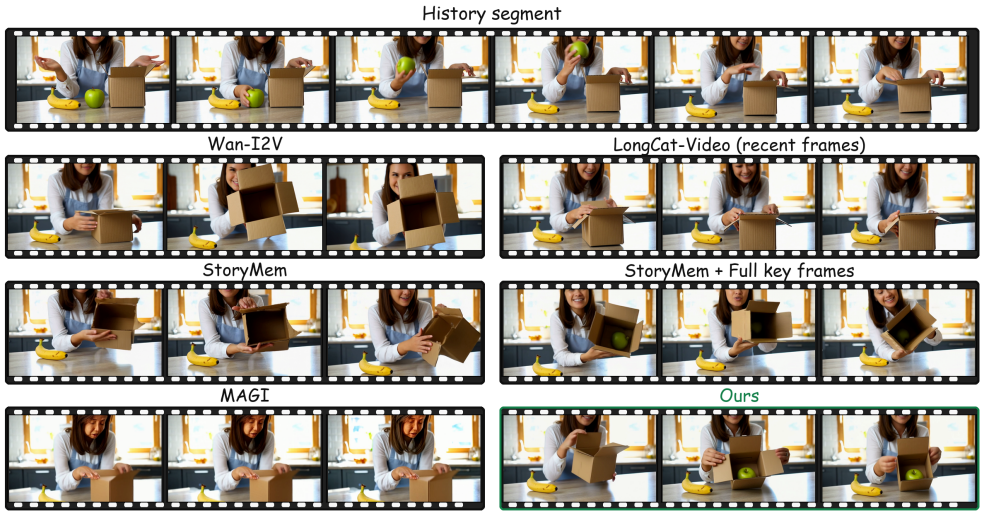

Figure 4: **Example on StateBench.** Setup prompt: a woman drops a green apple into an open cardboard box while a banana remains on the kitchen counter. Continuation prompt: "A single continuous realistic video on a kitchen counter. The woman tilts the cardboard box toward the camera to show what is inside. Fixed camera, medium shot, bright indoor lighting."

	Past-visible				Occluded-process				Complex-transition				Overall		
Method	SES↑	SC↑	HR↓	SCS↑	SES↑	SC↑	HR↓	SCS↑	SES↑	SC↑	HR↓	SCS↑	SES↑	SCS↑	SCS-All↑
Wan-I2V	69.4	26.0	40.7	42.9	90.8	22.6	49.2	36.1	76.0	17.1	18.4	48.2	78.0	41.6	32.5
LongCat-Video (recent frames)	57.6	27.9	22.4	51.8	70.8	23.9	27.2	46.0	60.0	10.6	**10.0**	49.2	62.5	49.0	30.6
StoryMem (Zhang et al., 2025)	80.0	50.5	15.4	66.9	75.4	34.7	20.4	54.8	66.0	27.8	34.8	46.5	75.0	58.5	43.9
StoryMem + full KFs	85.9	47.5	**11.0**	67.3	84.6	29.7	17.3	53.2	56.0	23.8	33.9	44.8	78.0	58.3	45.4
MAGI-1 (Teng et al., 2025)	40.0	19.6	23.5	47.1	40.0	21.8	51.9	33.5	12.0	44.4	25.0	59.2	33.0	42.8	14.1
StateAgent	**89.4**	**61.0**	11.2	**74.1**	**96.9**	**68.8**	**10.3**	**78.5**	**92.0**	**61.2**	17.4	**71.4**	**92.5**	**74.9**	**69.3**

Table 1: **Main StateBench results.** SES measures completion of the requested state-exposing or state-updating action; SCS denotes conditional state consistency on SES-positive samples. SC is the yes rate over state-correctness questions, while HR is the no rate over anti-hallucination questions. SCS-All is the all-case state score and treats exposure failure as failure.

5 Experiments

5.1 Experimental Setup

StateAgent uses Qwen3.5-Plus(Yang et al., 2025) as the parser for historical states, the state updater, and the editing-instruction generator, Wan2.7-Image(Mao et al., 2026) as the future-frame generator and Wan2.2-KF2V-Flash as the state-to-video renderer. We evaluate all methods under the same input (V_t, p_t, p_{t+1}). Each method generates only the next segment; the previous video is used only as conditioning evidence. Unless otherwise specified, output length, resolution, prompt template, sampling settings, and random seeds are matched within each model family. We report SES and SCS across the three StateBench categories. For diagnosis, we also report State Consistency (SC) and Hallucination Rate (HR) within each category; the Overall columns report aggregate SES, conditional SCS, and SCS-All.

Main baselines. The main comparison covers three families of continuation interfaces. *Frame-conditioned baselines* include I2V from the final frame and a recent-frame(Team et al., 2025) variant that conditions on the last few frames. *Key-frame and memory baselines* include StoryMem (Zhang et al., 2025) and StoryMem supplied with full key frames covering the complete state-establishing process. *Autoregressive baselines* include MAGI (Teng et al., 2025).

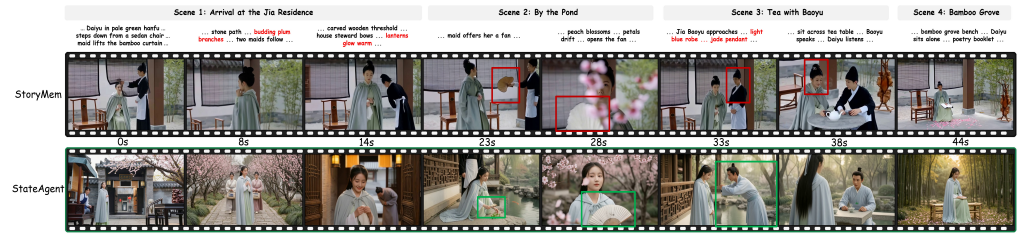

Figure 5: Long story generation example of ST-Bench. The same story script is given to StoryMem and STATEAGENT.

Variant	SES↑	SCS↑	SCS-All↑
Base I2V	78.0	41.6	32.5
Direct end-frame Editing	79.0	56.6	44.7
State Prompt Enhancement	90.0	61.9	55.8
Full STATEAGENT	**92.5**	**74.9**	**69.3**

Table 2: Ablation on state conditioning.

Method	Aesthetic	PF	Consistency
StoryMem	5.81	0.222	0.586
STATEAGENT	**6.35**	**0.230**	**0.594**

Table 3: ST-Bench results for multi-shot story generation.

5.2 MAIN RESULTS ON STATEBENCH

Table 1 evaluates whether stronger visual history interfaces let current generators recover the state needed for cross-segment handoff. Although observation-memory methods often complete the action described by the prompt, they frequently expose a state inconsistent with the previous segment. STATEAGENT improves conditional SCS from 58.5 to 74.9 over the strongest baseline, and improves the all-case SCS-All from 45.2 to 69.3, showing that the bottleneck is not merely access to historical evidence, but converting that evidence into the state required by the continuation.

The category-wise results reveal three failure modes. In Past-visible cases, retrieved frames help, but the model must still decide whether a past observation remains current. In Occluded-process cases, the target state is produced by a hidden update and never appears as a reusable target frame. In Complex-transition cases, similarity to past observations is a weak cue for spatial transfer, quantity change, color mixing, or physical-state change. Figure 4 makes these failures concrete: some baselines fail to tilt the box toward the camera, so the state-checking action is incomplete; others open the box but show an empty interior despite the apple being placed inside in the previous segment; still others hallucinate an ambiguous or mismatched object. STATEAGENT avoids these failures by predicting that the apple should remain inside the box and grounding that target state as a future end frame before rendering.

5.3 ABLATION ON STATE CONDITIONING

Table 2 shows that the gain is not from stronger hints alone. State-based prompt enhancement improves SES to 90.0, confirming that textual state hints help execute the requested action, but its SCS remains 61.9 because text lacks the visual grounding needed to bind the predicted state to entity appearance and so on. Direct end-frame editing provides such visual grounding and improves over several observation-memory baselines, but without the full state-update pipeline it still trails STATEAGENT. The full pipeline achieves the best SCS and SCS-All by coupling state prediction, future-frame grounding, and video rendering.

5.4 STATE CONSISTENCY IN MULTI-SHOT STORY GENERATION

To test whether our method remains useful beyond the controlled unit setting of STATEBENCH, we evaluate multi-shot story generation under the ST-Bench protocol from StoryMem (Zhang et al., 2025). ST-Bench consists of 1-minute long story scripts, shot-level video prompts, and scene-cut indicators. Following this protocol, each method generates the story shot by shot from the same script-level inputs. We evaluate aesthetic quality, prompt following, and cross-shot consistency following StoryMem.

Table 3 reports the ST-Bench metrics, while Figure 5 provides a qualitative comparison. STATEAGENT achieves improvements across all metrics. The qualitative example further shows that explicit state update helps preserve entity states and object appearances while following scene-level prompt changes.

The qualitative example reveals the limitation of a fixed visual memory bank. StoryMem stores only a limited number of memory frames, with early key frames acting as a memory sink and recent frames capturing short-term dynamics. This can make early characters and scenes repeatedly dominate later generations, causing repeated backgrounds, incorrect binding of new characters to earlier appearances, and loss of small but state-relevant objects. For example, a palm-leaf fan may drift into a paper fan when its appearance is diluted in memory. STATEAGENT instead maintains explicit entity states and updates them under each prompt, allowing it to follow scene changes, preserve object appearance and state, and avoid incorrect character binding.

6 RELATED WORK

Video continuation and visual memory. Video generators achieve strong short-term quality with DiT backbones (Ho et al., 2022b;a; Singer et al., 2022; Peebles & Xie, 2023; Ma et al., 2024; Wan et al., 2025; Kong et al., 2024). Long-video systems extend them through chunkwise continuation (Henschel et al., 2024), selected keyframes (Zhou et al., 2024; Zhang et al., 2025), entity or spatial memories (Zhou et al., 2026; Xiao et al., 2026; Li et al., 2025; Wu et al., 2026b), and KV caches (Chen et al., 2026b; Zhao et al., 2026; Meng et al., 2026). These interfaces preserve useful observations such as appearance, identity and so on. Our focus is complementary: converting historical observations into the current world state required by the next segment.

State evolution and video world models. Recent systems have started to explore evolving unseen states. PAN models long-horizon dynamics with LLM latent features and a video decoder (Xiang et al., 2025), but its state lacks explicit appearance binding for continuation. LiveWorld maintains a persistent 4D spatiotemporal memory for out-of-sight dynamics (Duan et al., 2026), but this powerful scene-level representation is costly. ReMind trains video generators to use KV-cache mechanisms as dynamic memory for hidden-state evolution (Xu et al., 2026), showing that specialized training can improve this ability. Our work instead evaluates this capability in current continuation systems and tests an explicit state carrier without retraining the video generator.

Benchmarks for generation, memory, and state reasoning. Video-generation benchmarks initially focused on visual quality, text alignment and so on (Huang et al., 2024; Liu et al., 2024; Sun et al., 2025; Bansal et al., 2024; Meng et al., 2024; Upadhyay et al., 2026). Recent benchmarks have begun to shift toward memory, state reasoning, and out-of-sight state evolution. WorldReasonBench evaluates whether video generators can predict future world states from an initial condition and action (Wu et al., 2026a). MIND studies memory consistency and action control in video world models (Ye et al., 2026). STEVO-Bench studies whether natural state evolution can continue when observation is interrupted by occluders, lights-off intervals, or look-away trajectories, but state setup, hiding, evolution, and reveal are coupled within one generation, making failures hard to attribute (Ma et al., 2026). MemoBench studies revisiting behavior in dynamically changing environments, but its focus is camera-controlled video generation (Chen et al., 2026a). Despite this progress, world-state handoff in cross-segment video continuation remains underexplored: a historical video establishes a world state, and a later prompt asks the next generated segment to use or change it.

7 CONCLUSION

We introduced cross-segment state handoff as a requirement distinct from observation memory in video continuation. We proposed STATEBENCH to test whether continuations preserve and update states established by previous segments across past-visible, occluded-process, and complex-transition cases. We also presented STATEAGENT, which explicitly predicts the continuation state, grounds it as a future end frame, and renders the next segment. Experiments show that stronger visual memory alone does not make current generators reliably infer world state; coherent continuation benefits from an explicit state belief that can be updated, grounded to appearance, and carried across generation boundaries, with supporting evidence from ST-Bench story generation.

REFERENCES

Hritik Bansal, Zongyu Lin, Tianyi Xie, Zeshun Zong, Michal Yarom, Yonatan Bitton, Chenfanfu Jiang, Yizhou Sun, Kai-Wei Chang, and Aditya Grover. Videophy: Evaluating physical commonsense for video generation, 2024.

Haoyu Chen, Kaichen Zhou, Hang Hua, Kaile Zhang, Jingwen Qian, Wufei Ma, Haonan Chen, Chunjiang Liu, Yizhou Zhao, Xiaoyuan Wang, et al. Memobench: Benchmarking world modeling in dynamically changing environments, 2026a.

Shuo Chen, Cong Wei, Sun Sun, Tiancheng Shen, Ping Nie, Kai Zou, Ge Zhang, Ming-Hsuan Yang, and Wenhu Chen. Context forcing: Consistent autoregressive video generation with long context. In *Forty-third International Conference on Machine Learning*, 2026b.

Zicheng Duan, Jiatong Xia, Zeyu Zhang, Wenbo Zhang, Gengze Zhou, Chenhui Gou, Yefei He, Feng Chen, Xinyu Zhang, and Lingqiao Liu. Liveworld: Simulating out-of-sight dynamics in generative video world models, 2026.

Roberto Henschel, Levon Khachatryan, Hayk Poghosyan, Daniil Hayrapetyan, Vahram Tadevosyan, Zhangyang Wang, Shant Navasardyan, and Humphrey Shi. Streamingt2v: Consistent, dynamic, and extendable long video generation from text, 2024.

Jonathan Ho, William Chan, Chitwan Saharia, Jay Whang, Ruiqi Gao, Alexey Gritsenko, Diederik P Kingma, Ben Poole, Mohammad Norouzi, David J Fleet, et al. Imagen video: High definition video generation with diffusion models, 2022a.

Jonathan Ho, Tim Salimans, Alexey A Gritsenko, William Chan, Mohammad Norouzi, and David J Fleet. Video diffusion models. In *ICLR workshop on deep generative models for highly structured data*, 2022b.

Ziqi Huang, Yinan He, Jiashuo Yu, Fan Zhang, Chenyang Si, Yuming Jiang, Yuanhan Zhang, Tianxing Wu, Qingyang Jin, Nattapol Chanpaisit, et al. Vbench: Comprehensive benchmark suite for video generative models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 21807–21818, 2024.

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

Runjia Li, Philip Torr, Andrea Vedaldi, and Tomas Jakab. Vmem: Consistent interactive video scene generation with surfel-indexed view memory. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 25690–25699, 2025.

Yaofang Liu, Xiaodong Cun, Xuebo Liu, Xintao Wang, Yong Zhang, Haoxin Chen, Yang Liu, Tieyong Zeng, Raymond Chan, and Ying Shan. Evalcrafter: Benchmarking and evaluating large video generation models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 22139–22149, 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, 2024.

Ziqi Ma, Mengzhan Liufu, and Georgia Gkioxari. Out of sight, out of mind? evaluating state evolution in video world models, 2026.

Chaojie Mao, Chen-Wei Xie, Chongyang Zhong, Haoyou Deng, Jiaxing Zhao, Jie Xiao, Jinbo Xing, Jingfeng Zhang, Jingren Zhou, Jingyi Zhang, Jun Dan, Kai Zhu, Kang Zhao, Keyu Yan, Minghui Chen, Pandeng Li, Shuangle Chen, Tong Shen, Yu Liu, Yue Jiang, Yulin Pan, Yuxiang Tuo, Zeyinzi Jiang, Zhen Han, Ang Wang, Bang Zhang, Baole Ai, Bin Wen, Boang Feng, Feiwu Yu, Gang Wang, Haiming Zhao, He Kang, Jianjing Xiang, Jianyuan Zeng, Jinkai Wang, Junjie Zhou, Ke Sun, Linqian Wu, Pei Gong, Pingyu Wu, Ruiwen Wu, Tongtong Su, Wenmeng Zhou, Wenting Shen, Wenyuan Yu, Xianjun Xu, Xiaoming Huang, Xiejie Shen, Xin Xu, Yan Kou, Yangyu Lv, Yifan Zhai, Yitong Huang, Yun Zheng, Yuntao Hong, Zhe Zhang, and Zhicheng Zhang. Wanimage: Pushing the boundaries of generative visual intelligence, 2026. URL https://arxiv.org/abs/2604.19858.

Fanqing Meng, Jiaqi Liao, Xinyu Tan, Wenqi Shao, Quanfeng Lu, Kaipeng Zhang, Yu Cheng, Dianqi Li, Yu Qiao, and Ping Luo. Towards world simulator: Crafting physical commonsense-based benchmark for video generation, 2024.

Yihao Meng, Zichen Liu, Hao Ouyang, Qiuyu Wang, Ka Leong Cheng, Yue Yu, Hanlin Wang, Haobo Li, Jiapeng Zhu, Yanhong Zeng, et al. Causalcine: Real-time autoregressive generation for multi-shot video narratives, 2026.

William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 4195–4205, 2023.

Uriel Singer, Adam Polyak, Thomas Hayes, Xi Yin, Jie An, Songyang Zhang, Qiyuan Hu, Harry Yang, Oron Ashual, Oran Gafni, et al. Make-a-video: Text-to-video generation without text-video data, 2022.

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 *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 8406–8416, 2025.

Meituan LongCat Team, Xunliang Cai, Qilong Huang, Zhuoliang Kang, Hongyu Li, Shijun Liang, Liya Ma, Siyu Ren, Xiaoming Wei, Rixu Xie, and Tong Zhang. Longcat-video technical report, 2025. URL https://arxiv.org/abs/2510.22200.

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

Rishi Upadhyay, Howard Zhang, Jim Solomon, Ayush Agrawal, Pranay Boreddy, Shruti Satya Narayana, Yunhao Ba, Alex Wong, Celso M de Melo, and Achuta Kadambi. Worldbench: Disambiguating physics for diagnostic evaluation of world models, 2026.

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, 2025.

Keming Wu, Yijing Cui, Wenhan Xue, Qijie Wang, Xuan Luo, Zhiyuan Feng, Zuhao Yang, Sudong Wang, Sicong Jiang, Haowei Zhu, et al. Worldreasonbench: Human-aligned stress testing of video generators as future world-state predictors, 2026a.

Tong Wu, Shuai Yang, Ryan Po, Yinghao Xu, Ziwei Liu, Dahua Lin, and Gordon Wetzstein. Video world models with long-term spatial memory. *Advances in Neural Information Processing Systems*, 38:49371–49393, 2026b.

Jiannan Xiang, Yi Gu, Zihan Liu, Zeyu Feng, Qiyue Gao, Yiyan Hu, Benhao Huang, Guangyi Liu, Yichi Yang, Kun Zhou, et al. Pan: A world model for general, interactable, and long-horizon world simulation, 2025.

Zeqi Xiao, Yushi Lan, Yifan Zhou, Wenqi Ouyang, Shuai Yang, Yanhong Zeng, and Xingang Pan. Worldmem: Long-term consistent world simulation with memory. *Advances in Neural Information Processing Systems*, 38:49632–49652, 2026.

Tianshuo Xu, Yichen Xie, Depu Meng, Chensheng Peng, Quentin Herau, Bo Jiang, Yihan Hu, and Wei Zhan. Teaching video generators to remember: Eliciting dynamic memory for out-of-sight state evolution, 2026.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, Chujie Zheng, Dayiheng Liu, Fan Zhou, Fei Huang, Feng Hu, Hao Ge, Haoran Wei, Huan Lin, Jialong Tang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jing Zhou, Jingren Zhou, Junyang Lin, Kai Dang, Keqin Bao, Kexin Yang, Le Yu, Lianghao Deng, Mei Li, Mingfeng Xue, Mingze Li, Pei Zhang, Peng Wang, Qin Zhu, Rui Men, Ruize Gao, Shixuan Liu, Shuang Luo, Tianhao Li, Tianyi Tang, Wenbiao Yin, Xingzhang Ren, Xinyu Wang, Xinyu Zhang, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yinger Zhang, Yu Wan, Yuqiong Liu, Zekun Wang, Zeyu Cui, Zhenru Zhang, Zhipeng Zhou, and Zihan Qiu. Qwen3 technical report, 2025. URL https://arxiv.org/abs/2505.09388.

Yixuan Ye, Xuanyu Lu, Yuxin Jiang, Yuchao Gu, Rui Zhao, Qiwei Liang, Jiachun Pan, Fengda Zhang, Weijia Wu, and Alex Jinpeng Wang. Mind: Benchmarking memory consistency and action control in world models, 2026.

Kaiwen Zhang, Liming Jiang, Angtian Wang, Jacob Zhiyuan Fang, Tiancheng Zhi, Qing Yan, Hao Kang, Xin Lu, and Xingang Pan. Storymem: Multi-shot long video storytelling with memory, 2025.

Yuyang Zhang, Wenyao Zhang, Zekun Qi, He Zhang, Haitao Lin, Jingbo Zhang, Yao Mu, Xiaokang Yang, Wenjun Zeng, and Xin Jin. Imagewam: Do world action models really need video generation, or just image editing?, 2026.

Min Zhao, Hongzhou Zhu, Kaiwen Zheng, Zihan Zhou, Bokai Yan, Xinyuan Li, Xiao Yang, Chongxuan Li, and Jun Zhu. Causal forcing++: Scalable few-step autoregressive diffusion distillation for real-time interactive video generation, 2026.

Jinsong Zhou, Yihua Du, Xinli Xu, Luozhou Wang, Zijie Zhuang, Yehang Zhang, Shuaibo Li, Xiaojun Hu, Bolan Su, and Ying-cong Chen. Videomemory: Toward consistent video generation via memory integration, 2026.

Yupeng Zhou, Daquan Zhou, Ming-Ming Cheng, Jiashi Feng, and Qibin Hou. Storydiffusion: Consistent self-attention for long-range image and video generation. *Advances in Neural Information Processing Systems*, 37:110315–110340, 2024.