EVOMA: EVOLVING ENTITY MEMORY AGENT FOR CONSISTENT LONG VIDEO GENERATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Consistent long video generation requires preserving entity identity while tracking evolving states. Existing planning-based methods typically predefine entity-state trajectories and visual references, which can diverge from the actual content of generated clips and continuing user inputs. We introduce EvoMA, an Evolving Entity Memory Agent that treats memory management as an online decision-making process. EvoMA maintains a structured memory that separates persistent identities from evolving states and records state histories for characters, scenes, and objects using textual descriptions and visual references. A vision-language agent updates and queries this memory through an OBSERVE–THINK–ACT–VERIFY loop, dynamically performing ADD, UPDATE, NOOP, and RETRIEVE operations based on newly generated clips. Memory changes are verified against visual evidence before being committed, while retrieved entity-state references guide subsequent generation. We further train the memory agent with reinforcement learning to improve both memory updates and state retrieval. Experiments show that EvoMA improves identity consistency and state alignment in long and interactive video generation.

1 INTRODUCTION

Long-form video generation is essential to applications such as filmmaking, advertising, storytelling, and interactive content creation, where coherence must be maintained across multiple shots. Recent advances in video generation models (Brooks et al., 2024; OpenAI, 2025; Google DeepMind, 2025; Team Seedance et al., 2026; HaCohen et al., 2026; Tencent Hunyuan Foundation Model Team, 2025; Wan Team et al., 2025) and agentic generation frameworks (Lin et al., 2023; Hu et al., 2024; Wu et al., 2025) have made such workflows increasingly feasible through iterative planning, generation, and revision. Yet as generation unfolds, entities may evolve due to both generated content and subsequent user edits (e.g., a character may change clothes, an object may be moved or altered, or a scene may be revised), while their identities and previously established states must remain consistent when they reappear. In this work, we study how to dynamically maintain and retrieve evolving entity states so that each new shot remains consistent with prior generated content and user interactions.

This problem is challenging for several reasons. First, the generated state is only partially specified by the initial instruction or plan, with many details determined during generation; memory must therefore be grounded in the generated content rather than the planned trajectory alone. Second, not every visual variation should be remembered: transient actions, viewpoint changes, occlusions, and generation artifacts must be distinguished from persistent state changes. Third, an entity may need to revisit an earlier state rather than simply continue from its latest appearance, requiring selective retrieval over its state history. Errors in either memory writing or retrieval can then propagate across subsequent shots and accumulate over time.

Existing long-video generation methods (Yang et al., 2025; Zhang et al., 2025; Yi et al., 2025; Cai et al., 2025) address temporal consistency mainly by propagating historical context through previous frames, latent caches, keyframes, or predefined visual references. More recent agentic approaches (Lin et al., 2023; Hu et al., 2024; Wu et al., 2025; Huang et al., 2026) further introduce structured storyboards, character banks, and memory modules to coordinate generation across shots. However, these mechanisms largely treat memory as either a fixed plan or a collection of histori-

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: **Dynamic evolving memory in EvoMA.** Compared with a fixed, pre-planned reference bank, EvoMA maintains an *evolving* memory bank whose entity states are dynamically updated during generation (WRITE), enabling flexible state mutation and selective retrieval of historical states (RETRIEVE).

cal observations, rather than an explicit record of how individual entities evolve over time. These limitations call for explicit decisions about which entity states to preserve and retrieve.

In this paper, we formulate long-video memory management as an online decision-making problem over evolving entity states. Our central insight is to separate persistent entity identity from evolving visual states, and to update memory according to generated visual evidence and subsequent user inputs rather than a predefined trajectory. Based on this perspective, we introduce EvoMA, an Evolving Entity Memory Agent that maintains reusable state histories for recurring characters, scenes, and objects (Figure 1). After each generated clip, a vision-language agent decides whether to ADD, UPDATE, or preserve memory via NOOP; before the next clip, it RETRIEVEs the entity states most relevant to the current request, enabling earlier states to be revisited when needed.

Reliable memory management further requires distinguishing persistent state changes from transient visual variations and deciding when memory should remain unchanged. EvoMA addresses this with an OBSERVE–THINK–ACT–VERIFY loop that grounds candidate updates in generated visual evidence and verifies them against existing memory before commitment. We further optimize dedicated writing and retrieval policies with reinforcement learning, using structured credit assignment to improve both state updates and state selection.

We evaluate EvoMA on long-horizon and interactive multi-shot video generation. Experiments show that dynamically grounded memory improves entity persistence and temporal consistency over static or pre-planned memory, while ablations validate the contributions of grounded verification, policy optimization, and the global planner.

Our main contributions are:

- We formulate memory management for long-video generation as an online entity-state decision problem, in which memory is continuously grounded in the realized visual trajectory rather than fixed by a pre-generated plan.
- We introduce a dynamic entity-centric memory representation and agentic management loop that separates persistent identities from mutable states and supports explicit ADD, UPDATE, NOOP, and RETRIEVE operations with visual verification.
- We develop reinforcement-learned writing and retrieval policies for deciding when and how memory should evolve and which historical states should condition future generation.
- We demonstrate that the resulting system improves consistency in long-horizon and interactive video generation, particularly when entities evolve or previously established states must be reused.

2 Related Work

2.1 Long-horizon and Agentic Video Generation

Multi-shot video generation aims to synthesize coherent content across discontinuous yet semantically connected shots. Early visual-storytelling pipelines combine subject-consistent keyframes with motion prediction (Zhou et al., 2024), while recent methods mainly follow two directions: backbone-level context extension (Guo et al., 2025; Qi et al., 2025; Meng et al., 2026) and storyboard-driven visual planning (Zheng et al., 2024; An et al., 2026; Zhang et al., 2026; Liu et al., 2026). The former propagates temporal context through inter-shot attention or segment-level alignment, whereas the latter conditions generation on keyframes, transition pairs, or historical frames.

More recently, agentic frameworks formulate multi-shot generation as an iterative production process, using LLMs to decompose scripts, assign specialized roles, invoke generation tools, and refine outputs through verification (Lin et al., 2023; Zhuang et al., 2024; Xie et al., 2024; Hu et al., 2024; Wu et al., 2025; Wang et al., 2026; Xie et al., 2026; Liang et al., 2025; Tang et al., 2025; Song et al., 2026). These approaches improve cross-shot coordination and controllability, but their persistent context remains dominated by visual references, storyboards, character sheets, or other static artifacts, rather than an explicit model of how entity states evolve throughout generation.

2.2 Memory Mechanisms in Video Generation

Memory in long video generation is commonly maintained through token- or latent-level context, including streaming caches, sparse routing, and historical latent retrieval (Yang et al., 2025; Yi et al., 2025; Cai et al., 2025; Hu et al., 2026), or through frame-level keyframes and visual reference buffers (Zhang et al., 2025; Huang et al., 2026). Related world-model approaches retrieve historical latents using continuous camera trajectories (Yu et al., 2025a; Xiao et al., 2026). These mechanisms largely represent memory as feature caches or visual snapshots, without explicitly modeling entity-state evolution across discontinuous shots.

Recent methods move toward entity-centric memory, but either rely on pre-planned storyboards (Zhou et al., 2026) or update memory from generated frames without explicit state-transition modeling (Yin et al., 2026). In contrast, EvoMA represents memory as structured entity-state sequences and manages them through explicit ADD, UPDATE, NOOP, and RETRIEVE operations, allowing memory to evolve with generated content while preserving reusable historical states.

3 Method

EvoMA (Figure 2) formulates long-video memory management as an online decision process over evolving entity states. The framework integrates a structured entity–state representation (Section 3.1), an evidence-grounded loop (Section 3.2), and reinforcement-learned writing and retrieval policies (Section 3.3).

3.1 Dynamic Entity-Centric Memory

Persistent Identities and Reusable States. To maintain cross-clip consistency, we decouple an entity's persistent identity from its temporal visual states. An entity e represents a recurring character, scene, or object, while a state captures its specific visual configuration. After processing clip t, the memory bank is structured as:

$$\mathcal{M}_t = \{m_e\}_{e \in \mathcal{E}_t}, \qquad m_e = \big(\mathrm{id}_e, c_e, d_e, \mathcal{S}_e\big), \tag{1}$$

where id_e is a unique key, $c_e \in \{\mathrm{character}, \mathrm{scene}, \mathrm{object}\}$ denotes the category, d_e is a canonical identity description serving as the global semantic reference, and $\mathcal{S}_e$ is the catalog of instantiated visual states.

Each state $s_{e,k} = (k, \delta_{e,k}, \mathcal{I}_{e,k}) \in \mathcal{S}_e$ comprises a local index k, a textual description $\delta_{e,k}$ of the resulting state's mutable attributes (e.g., clothing, accessories, lighting), and synthesized reference images $\mathcal{I}_{e,k}$. A transition produces $s_{e,k}$ by applying attribute changes Δ_k to an existing base state $s_{e,k_{\mathrm{base}}}$, preserving the entity's identity and all unaffected attributes.

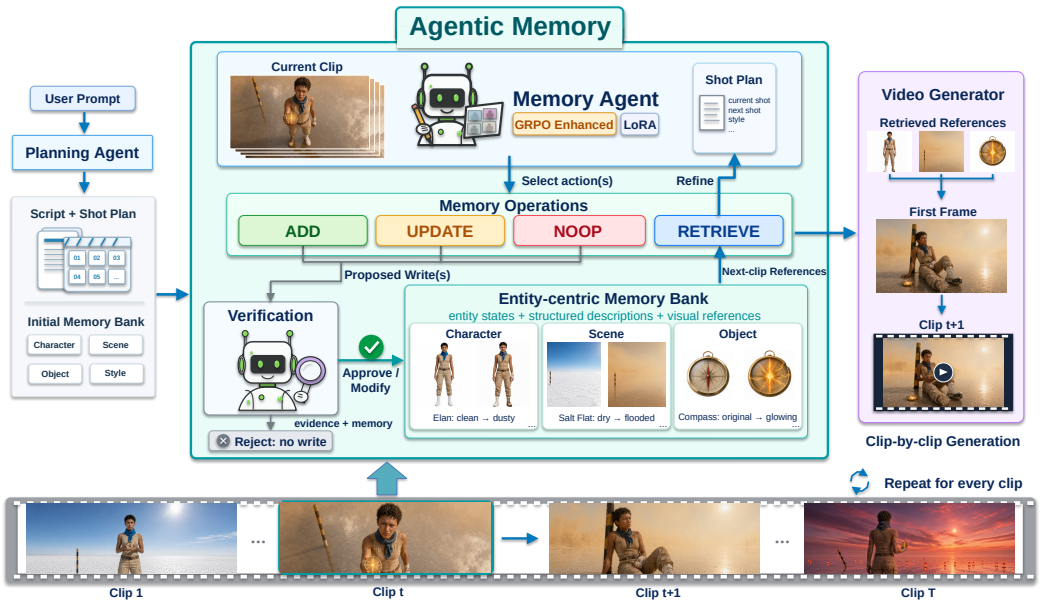

Figure 2: **Overview of EvoMA.** Between consecutive clips, EvoMA deliberates over visual evidence to select memory actions (ADD, UPDATE, NOOP), verifies proposed modifications before committing them to an entity-centric bank, and retrieves historical references to condition subsequent generation. Policies are optimized via GRPO.

Disentangled Visual Reference Synthesis. Storing raw video frames directly entangles foreground entities with ambient backgrounds, leading to visual artifacts during cross-scene transitions. We thus synthesize decoupled visual references conditioned on identity d_e, dynamic features $\delta_{e,k}$, and available visual evidence. Specifically, character and object references are rendered against clean, neutral backgrounds, whereas scene references capture environmental context without foreground subjects. Initial states seed $\mathcal{M}_0$, while new states are dynamically extracted from generated clips. Global visual style is governed separately via unified textual conditioning.

Memory Writing. After clip generation, the writing policy selects one of three mutually exclusive memory actions for each candidate entity; proposed modifications are verified before being committed (Section 3.2): ADD registers an unseen entity along with its initial state; UPDATE records an evolved appearance by appending a new state to the entity's catalog $\mathcal{S}_e$ while preserving its persistent identity $(\mathrm{id}_e, c_e, d_e)$; and NOOP keeps memory intact when observed variations are transient or already faithfully captured.

State Retrieval. Prior to generating shot t, RETRIEVE queries memory $\mathcal{M}_{t-1}$ to select relevant historical appearances without altering the bank. For each relevant entity e, it selects at most one historical state $s_{e,k} \in \mathcal{S}_e$, binding the entity's canonical description d_e, dynamic attributes $\delta_{e,k}$, and visual reference $\mathcal{I}_{e,k}$ to shot prompt u_t. This enables EvoMA to recall and reuse historical appearances across discontinuous shots rather than merely propagating the latest frame.

3.2 EVIDENCE-GROUNDED LOOP

To prevent ephemeral artifacts or rendering flicker from corrupting memory, EvoMA regulates state evolution between consecutive shots through an OBSERVE–THINK–ACT–VERIFY loop:

$$
\begin{aligned}
o_t &= \mathrm{Observe}(v_t), && \text{[Observe]} \\
(\tau_t, a_t) &\sim \pi_\theta^{\mathrm{write}}(\cdot \mid x_t), && \text{[Think \& Act]} \\
\mathcal{M}_t &= \mathrm{Verify}(a_t, \tau_t, o_t, \mathcal{M}_{t-1}), && \text{[Verify]}
\end{aligned} \tag{2}
$$

where decision context $x_t = (o_t, \mathcal{M}_{t-1}, \mathcal{R}_t, u_t)$ unifies observation evidence o_t, prior memory $\mathcal{M}_{t-1}$, active references $\mathcal{R}_t$, and current shot request u_t.

Observe: Evidence Extraction. To ground transitions in physical video content rather than speculative textual intent, the agent extracts representative frames from clip v_t alongside active conditioning references $\mathcal{R}_t$. This forms observation o_t, anchoring identity tracking in verified visual evidence.

Think: State Deliberation. The writing policy $\pi_\theta^{\text{write}}$ produces a reasoning trace τ_t over o_t to distinguish persistent state transitions (e.g., costume changes or physical injury) from transient shot-level variations (e.g., momentary character poses or rendering flicker). It determines whether observed variations instantiate a new entity, derive from an existing base state $s_{e,k_{\text{base}}} \in \mathcal{S}_e$ via attribute mutation Δ_k, or represent negligible drift that should be ignored.

Act: Candidate Formulation. Conditioned on deliberation τ_t, the policy formulates a structured candidate action a_t. For each entity category $c_e \in \mathcal{C}$, the action specifies an explicit decision g_c alongside candidate records specifying target identifier id_e, attribute delta Δ_k, base state index k_{base}, and supporting timestamps. Crucially, a_t operates in an isolated proposal buffer; persistent memory $\mathcal{M}_{t-1}$ remains immutable until verification succeeds.

Verify: Multimodal Grounded Verification. To safeguard memory integrity, Verify audits candidate action a_t and rationale τ_t against observation o_t and historical memory $\mathcal{M}_{t-1}$. The verifier yields APPROVE, MODIFY (recalibrating base states or attribute text), or REJECT. Upon approval, decoupled reference images $\mathcal{I}_{e,k}$ are rendered, and verified mutations are committed into bank $\mathcal{M}_t$, ready for subsequent retrieval.

3.3 Enhancing the Memory Agent via Reinforcement Learning

Following the separation of memory management and utilization in Memory-R1 (Yan et al., 2026), we decouple the memory agent into two staged policies over multimodal backbones: a writing policy $\pi_\theta^{\text{write}}$ and a retrieval policy π_ϕ^{ret}, optimized sequentially.

Single-Shot Localization and Image-Free Rollout. To reduce compounding errors and simplify credit assignment over long horizons, policy updates are localized to *single-shot transitions* ($t \to t+1$) mined from narrative video trajectories. We first optimize the writing policy $\pi_\theta^{\text{write}}$ on decision contexts $x_t = (o_t, \mathcal{M}_{t-1}, \mathcal{R}_t, u_t)$ directly in structured text-attribute space on a logical memory bank, deliberately foregoing image generation to avoid synthesis noise and judging latency. Once converged, the optimal writer $\pi_{\theta^\star}^{\text{write}}$ is frozen to materialize student memory banks $\widetilde{\mathcal{M}}_t$ offline across teacher trajectories. The retrieval policy π_ϕ^{ret} is subsequently trained on these materialized contexts $(u_{t+1}, \widetilde{\mathcal{M}}_t)$, directly exposing π_ϕ^{ret} to realistic student error distributions. Trajectory mining details appear in Appendix A.1.

Two-Level Decision Structure and Structured Rewards. EvoMA represents policy outputs as $y = [(g_c, p_c)]_{c\in\mathcal{C}}$, where $\mathcal{C} = \{\text{character}, \text{scene}, \text{object}\}$, g_c is an arbitration gate, and p_c is a realization payload. Gates are NOOP/WRITE for writing and EMPTY/SELECT for retrieval; active payloads specify entity-level ADD/UPDATE operations or selected references, respectively. The 3 categories and 2 decision levels yield 6 decoupled optimization units $u = (c, \ell)$, with $\ell \in \{\text{arb}, \text{real}\}$. Using curated teacher trajectories, rewards for arbitration and realization are defined as:

$$r_c^{\text{arb}} = 2\mathbf{1}[g_c = g_c^\star] - 1, \qquad r_c^{\text{real}} = 2\mathbf{1}[g_c^\star = g^+ \ \wedge \ Q_c = 1] - 1, \tag{3}$$

where $g^+ \in \{\text{WRITE}, \text{SELECT}\}$ signifies an active operation target, and binary indicator $Q_c \in \{0, 1\}$ evaluates realization correctness. For memory writing, $Q_c = Q_c^{\text{write}}$ jointly checks schema compliance, non-duplication, downstream utility, and semantic consistency. For state retrieval, $Q_c = Q_c^{\text{ret}}$ checks exact agreement between the selected reference set and the teacher target set after target reachability alignment (Appendix A.2). Both policies are optimized via Group Relative Policy Optimization (GRPO) (Shao et al., 2024), standardizing advantages independently across the decoupled units u with gradients restricted to semantic value tokens (see Appendix A.3 for the full objective and dynamic sampling, and Appendix B for training hyperparameters).

Table 1: Quantitative comparison on ST-Bench. We report Aesthetic Quality, Prompt Following, and Cross-shot Consistency ($[0, 1]$, higher is better).

		Prompt Following		Cross-shot Consistency	
Method	Aesthetic	Global	Single-shot	Overall	Top-10
Wan2.2-T2V	0.6452	0.2174	**0.2452**	0.3937	0.4248
StoryDiffusion + Wan2.2-I2V	0.6085	0.2288	0.2349	0.4600	0.4732
IC-LoRA + Wan2.2-I2V	0.5704	0.2131	0.2181	0.4110	0.4296
HoloCine	0.5653	0.2199	0.2125	0.4628	0.5005
StoryMem	0.6133	0.2289	0.2313	0.5065	0.5337
EvoMA (GRPO) + Wan2.2-I2V	**0.6714**	**0.2420**	0.2407	**0.5357**	**0.5595**

4 EXPERIMENTS

4.1 EXPERIMENTAL SETUP

Implementation Details. We instantiate EvoMA with Qwen3-VL-8B-Instruct (Bai et al., 2025) as the unified backbone for memory writing ($\pi_\theta^{\text{write}}$), visual verification (Verify), and state retrieval (π_ϕ^{ret}). While EvoMA (Base) operates zero-shot, EvoMA (GRPO) equips the writer and retriever with LoRA (Hu et al., 2021) adapters trained via GRPO with two-level credit assignment (Section 3.3). For image generation, we employ GPT Image 2.5 Flare (OpenAI, 2026). For video generation on EvoVideo-Long, MiniMax H3 (MiniMax, 2026) produces 960×544 clips via image-to-video synthesis; for ST-Bench and EvoVideo-Interactive, we standardize on Wan2.2 and Wan2.1, respectively. Appendix B details full training and inference parameters.

Evaluation Benchmarks. We evaluate EvoMA across both an established community benchmark, ST-Bench (Zhang et al., 2025) (30 multi-shot narrative stories, 300 video prompts), and our proposed benchmark, **EvoVideoBench**. Crucially, ST-Bench cannot validate dynamic memory tracking: its narratives are largely static, devoid of physical state mutations, and its coarse embedding metrics (e.g., ViCLIP (Wang et al., 2024)) fail to capture fine-grained transitions, inadvertently rewarding visual stagnation over legitimate state evolution. To address these limitations, we introduce **EvoVideoBench** to rigorously evaluate entity consistency and dynamic state fidelity across two complementary suites. (1) ***EvoVideo-Long.*** Spanning 32 cases and 384 shots across 8-shot (8 cases), 12-shot (16 cases), and 16-shot (8 cases) multi-shot narratives. It tracks 141 unique entities exhibiting non-monotonic physical and semantic transitions (e.g., costume changes, progressive injuries, tool degradation, and environmental illumination). (2) ***EvoVideo-Interactive.*** Spanning 20 cases and 240 shots simulating streaming video production with 60 sequential user interventions (2–4 edits per case at designated shot boundaries), introducing mid-stream additions, removals, attribute alterations, and explicit corrections without retroactively modifying prior clips. Initial visual states are grounded in verified reference bundles. Complete inventories appear in Appendix C.

Evaluation Metrics. On ST-Bench, we follow standard protocol (Zhang et al., 2025) reporting LAION aesthetic quality (Huang et al., 2024b), along with ViCLIP (Wang et al., 2024) for prompt following and cross-shot consistency. On EvoVideoBench, however, evaluating dynamic state transitions requires disentangling immutable identity from evolving state conditions: true consistency demands exhibiting the *correct, intended state* across time while preserving persistent identity. We evaluate shot-initial frames along two orthogonal axes: (1) ***Identity Consistency (I).*** Determines whether the depicted subject is the same target entity across shots by verifying its invariant identity against its earliest video anchor. (2) ***State Alignment (S).*** Determines whether the entity's visual state is correct by assessing compliance with current script-mandated conditions, independently of past frames. A frozen multimodal judge (Qwen3.7-Plus (Qwen Team, 2026), $T = 0$) conducts localized binary evaluations. For offline narrative generation on EvoVideo-Long (Table 2), macro-averaging yields the composite *Identity–State Alignment* (ISA) as the mean of I and S per category ($\text{ISA}_c = (I_c + S_c)/2$) and averaged across $\mathcal{C}$ (complete formulation in Appendix D). We also evaluate retrieved reference images using the same ISA aggregation, with identity assessed against the canonical initial reference bundle. For interactive streaming on EvoVideo-Interactive (Table 3), evaluation focuses on dynamic edited states via State Alignment (S). In addition to category-level

Table 2: Quantitative results on EvoVideo-Long. We report ISA on shot-initial frames and reference images ($[0, 1]$, higher is better).

	First Frame ISA				Reference Image ISA			
Method	Character	Scene	Object	Mean	Character	Scene	Object	Mean
AniMaker	0.2812	0.3858	0.4022	0.3564	N/A	N/A	N/A	N/A
MovieAgent	0.6768	0.4075	0.4033	0.4958	N/A	N/A	N/A	N/A
ViMax	0.6430	0.6125	0.5488	0.6015	N/A	N/A	N/A	N/A
VideoMemory	0.5950	0.7435	0.5001	0.6129	0.5862	0.7326	0.3059	0.5416
EvoMA (Base)	0.6747	0.8532	0.7036	0.7438	0.7380	0.8151	0.6679	0.7403
EvoMA (GRPO)	**0.7292**	**0.8698**	**0.7843**	**0.7944**	**0.7951**	**0.8701**	**0.8053**	**0.8235**

Table 3: Quantitative results on EvoVideo-Interactive. We report State Alignment (S), S_{onset}, and S_{persist} on shot-initial frames ($[0, 1]$, higher is better).

Method	Character S	Scene S	Object S	Mean S	S_{onset}	S_{persist}
LongLive 1.0	0.1507	0.0417	0.0886	0.0937	0.0663	0.1043
AniMaker	0.3496	0.6531	0.5526	0.5185	0.9321	0.3567
MovieAgent	0.1940	0.5422	0.4140	0.3834	**0.9534**	0.1605
ViMax	0.2032	0.4094	0.4667	0.3597	0.9139	0.1430
EvoMA (Base)	**0.9564**	0.9281	0.8553	0.9133	0.8617	**0.9334**
EvoMA (GRPO)	0.9490	**0.9417**	**0.8737**	**0.9214**	0.9238	0.9205

accuracy, we evaluate **Onset State Compliance** (S_{onset}), which measures whether an edit is immediately realized on its onset shot, and **Temporal State Persistence** (S_{persist}), which measures whether the edited state persists across subsequent unprompted shots. Protocols are detailed in Appendices D and F.

Baselines and Streaming Protocol. On ST-Bench, we compare against five story synthesis and personalization baselines: Wan2.2-T2V (Wan Team et al., 2025), StoryDiffusion (Zhou et al., 2024), IC-LoRA (Huang et al., 2024a), HoloCine (Meng et al., 2026), and StoryMem (Zhang et al., 2025). On EvoVideoBench, we benchmark EvoMA against four representative long-video systems: AniMaker (Shi et al., 2025), MovieAgent (Wu et al., 2025), ViMax (Huang et al., 2026), and VideoMemory (Zhou et al., 2026), alongside unoptimized EvoMA (Base). All systems share identical image (GPT Image 2.5 Flare) and video (MiniMax H3) generators on EvoVideo-Long, with Gemini-3.6-Flash (Google DeepMind, 2026b) serving as the global planner; VideoMemory's memory agent uses unoptimized Qwen3-VL-8B-Instruct (Bai et al., 2025), matching our base backbone. For interactive generation on EvoVideo-Interactive, we additionally benchmark against the continuous streaming baseline LongLive 1.0 (Yang et al., 2025). To enforce strict causal fairness, interactive methods are evaluated by exposing only the current user instruction and causally generated history at each shot boundary, strictly masking future edits and frames, and standardized on Wan2.1 (Wan Team et al., 2025) as the shared video generator. VideoMemory is omitted from interactive evaluation because its memory bank fundamentally requires full-storyboard global planning (Appendix F).

4.2 Main Results

Results on ST-Bench. As shown in Table 1, EvoMA achieves state-of-the-art aesthetic quality (0.6714), global prompt following (0.2420), and cross-shot consistency (0.5357). While unconstrained Wan2.2-T2V (Wan Team et al., 2025) slightly leads in single-shot text alignment, it suffers a severe drop in cross-shot coherence (0.3937 vs. 0.5357), confirming EvoMA's effectiveness on conventional static storyboards.

Long-Horizon Narrative Generation. As summarized in Table 2, EvoMA (GRPO) achieves the highest consistency across all entity categories on EvoVideo-Long, attaining 0.7944 ISA on shot-initial frames and 0.8235 on reference images. Decomposing metrics into Identity (I) and State (S) axes (Appendix Table 8) reveals that our primary advantage stems from dynamic state align-

Table 4: Human evaluation of consistency. We report pairwise win rates (%, higher is better) from 21 participants and 105 comparisons. Overall consistency is rated separately.

Consistency	AniMaker	MovieAgent	ViMax	VideoMemory	EvoMA (GRPO)
Character	26.2	42.9	23.8	28.6	**45.2**
Scene	19.0	38.1	26.2	56.1	**58.5**
Object	12.2	54.8	19.0	33.3	**56.1**
Overall	14.6	51.2	28.6	53.7	**70.7**



Figure 3: **Narrative state evolution on EvoVideo-Long.** Green (✓) and red (×) denote compliance and violation in tracking simultaneous entity mutations (signal color progression and character knee tape). While baselines exhibit premature or omitted transitions, EvoMA (GRPO) faithfully coordinates all state changes.

ment, reducing the use of outdated states across transitions: baselines like MovieAgent maintain identity by freezing states ($I_C = 0.9692, S_C = 0.3844$), whereas EvoMA elevates state alignment ($S = 0.6727, +0.0772$ over Base) while preserving identity. Parallel gains on reference images confirm that policy optimization directly enhances the retrieval of timely visual conditioning to steer generation.

Interactive Video Generation. Table 3 highlights how memory design governs interactive consistency. Continuous streaming model LongLive fails to reliably execute immediate user edits ($S_{\text{onset}} = 0.0663$). In contrast, while AniMaker, MovieAgent, and ViMax readily realize onset edits ($S_{\text{onset}} \geq 0.9139$), their lack of dynamic memory causes temporal persistence to collapse once edit prompts subside ($S_{\text{persist}} \leq 0.3567$). EvoMA resolves this dichotomy, sustaining high persistence ($S_{\text{persist}} \geq 0.9205$). Because EvoVideo-Interactive features sparser transitions and policies were trained offline without interactive rollout tuning, EvoMA (GRPO) performs comparably to Base (0.9214 vs. 0.9133 Mean S). Identity and ISA results appear in Appendix E.4.

Qualitative Analysis. Qualitative results in Figures 3 and 4 corroborate our quantitative findings. While baselines struggle with omitted or premature transitions during narrative progression (Fig-



Figure 4: **Interactive state persistence on EvoVideo-Interactive.** Following an edit request, green (✓) and red (×) denote success and failure/reversion. While baselines fail at onset or revert in later shots, EvoMA (GRPO) faithfully retains the edited state throughout.

ure 3) and suffer from rapid state reversion following interactive edits (Figure 4), EvoMA faithfully tracks concurrent entity mutations and sustains user interventions across extended horizons.

User Study. We conduct a blinded, counterbalanced pairwise study with 21 participants (105 comparisons across 8 cases), evaluating character, scene, object, and overall consistency. As shown in Table 4, EvoMA (GRPO) achieves the highest observed win rates across all four dimensions. Its overall-consistency win rate is 70.7%, versus VideoMemory's 53.7% (+17.0 percentage points), with both rates aggregated across each method's comparisons against all other methods. Appendix H details the viewing protocol and win-rate definition.

Ablation Studies. On the matched eight-case subset, sequentially optimizing both policies achieves 0.8176 first-frame ISA, compared with 0.7746 and 0.7738 when optimizing only the writer or retriever. GRPO also outperforms SFT (0.7641), which commits fewer memory states. Verification improves mean first-frame ISA for both Base and GRPO, and the benefit of policy optimization also holds with the alternative Qwen planner. Comprehensive metrics, breakdowns, and memory dynamics are detailed in Appendix G.

5 Conclusion

In this paper, we introduced EvoMA, an Evolving Entity Memory Agent that formulates memory management for long-horizon and interactive video generation as an online decision-making process over dynamic entity states. Central to our approach is the explicit disentanglement of persistent entity identities from mutable, reusable visual states across characters, scenes, and objects. By regulating state transitions through an evidence-grounded OBSERVE–THINK–ACT–VERIFY loop, EvoMA commits only verified visual mutations, reducing the risk that transient artifacts or unsupported observations are incorporated into persistent memory. Furthermore, we optimized the writing and retrieval policies sequentially via GRPO with a two-level decision structure that separates action arbitration from state realization, enabling the agent to learn when to update memory and which historical states to recall. Extensive evaluations on the EvoVideoBench benchmark demonstrate that EvoMA markedly surpasses existing static and planning-based baselines, harmonizing persistent identity retention with accurate dynamic state alignment over extended narrative arcs.

AI USE STATEMENT

We used generative AI tools to assist with writing and language polishing, draft portions of the manuscript, and generate synthetic datasets. Specifically, AI models were used to generate the narrative scripts for EvoVideo-Long. We also used Gemini 3.5 Flash to run our video-generation pipeline and collected the resulting execution traces as training data for our memory writing and retrieval policies. The models and data construction, filtering, and verification procedures are described in Section 4.1 and Appendices A.1 and C. The authors take full responsibility for the final content of this work, including all text, claims, and artifacts produced with the assistance of generative AI.

ETHICS STATEMENT

EvoMA aims to support creative video production by improving identity consistency and tracking evolving entity states. These capabilities could also facilitate convincing fabricated narratives, impersonation, and non-consensual depictions. Persistent memory may retain sensitive information in textual descriptions and visual references, while erroneous or stereotypical representations inherited from the underlying models may propagate across subsequent shots. Our visual verification mechanism checks consistency with generated content; it does not establish real-world truth, consent, or content safety. Responsible deployment should therefore include disclosure of synthetic content, safeguards against harmful requests, consent for the use of identifiable individuals, and controls over memory access, retention, and deletion.

Our evaluation focuses on narrative consistency and state alignment, rather than demographic fairness or safety, and improvements on these metrics should not be interpreted as evidence of either. The human evaluation involves 21 participants judging generated visual content, with aggregate results reported in the paper and the rating protocol described in Appendix H. Participant identifiers should be excluded from any public release of study responses. Any release of scripts, generated media, model adaptations, or execution traces should also undergo review for sensitive content and third-party rights, with data provenance and applicable usage restrictions documented. The use of synthetic training data does not by itself eliminate privacy, bias, or intellectual-property concerns.

REPRODUCIBILITY STATEMENT

To support reproducibility, Section 3 describes the memory representation, evidence-grounded control loop, and policy optimization framework. Appendix A details training-data construction and filtering, reward definitions, and the optimization objective. Section 4.1 and Appendix B document the model configurations, training hyperparameters, hardware, checkpoint selection, and memory-agent inference settings, while Appendix I provides the system prompts for memory writing, verification, and retrieval. Benchmark construction and case inventories are provided in Appendix C; scoring and aggregation rules, the ST-Bench setup, and the interactive evaluation protocol are described in Appendices D, E.5, and F, respectively. Appendix H specifies the human-evaluation interface, response options, and win-rate calculation. These descriptions support replication of the experimental procedure, although exact outputs may vary with stochastic generation and changes to externally hosted models.

REFERENCES

Zhaochong An, Menglin Jia, Haonan Qiu, Zijian Zhou, Xiaoke Huang, Zhiheng Liu, Weiming Ren, Kumara Kahatapitiya, Ding Liu, Sen He, et al. Onestory: Coherent multi-shot video generation with adaptive memory. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 16173–16184, 2026.

Shuai Bai, Yuxuan Cai, Ruizhe Chen, Keqin Chen, Xionghui Chen, Zesen Cheng, Lianghao Deng, Wei Ding, Chang Gao, Chunjiang Ge, et al. Qwen3-VL technical report. *arXiv preprint arXiv:2511.21631*, 2025.

Tim Brooks, Bill Peebles, Connor Holmes, Will DePue, Yufei Guo, Li Jing, David Schnurr, Joe Taylor, Troy Luhman, Eric Luhman, Clarence Ng, Ricky Wang, and Aditya Ramesh. Video generation models as world simulators. 2024. URL https://openai.com/index/video-generation-models-as-world-simulators/.

Shengqu Cai, Ceyuan Yang, Lvmin Zhang, Yuwei Guo, Junfei Xiao, Ziyan Yang, Yinghao Xu, Zhenheng Yang, Alan Yuille, Leonidas Guibas, et al. Mixture of contexts for long video generation. *arXiv preprint arXiv:2508.21058*, 2025.

Google DeepMind. Veo: A text-to-video generation system. Technical report, Google DeepMind, 2025. URL https://storage.googleapis.com/deepmind-media/veo/Veo-3-Tech-Report.pdf. Veo 3 Technical Report.

Google DeepMind. Gemini 3.5 Flash: Model card, May 2026a. URL https://deepmind.google/models/model-cards/gemini-3-5-flash/. Published May 19, 2026.

Google DeepMind. Gemini 3.6 Flash: Model card, July 2026b. URL https://deepmind.google/models/model-cards/gemini-3-6-flash/. Published July 2026.

Google DeepMind. Gemini 3.8 Flash: Model card, September 2026c. URL https://deepmind.google/models/model-cards/gemini-3-8-flash/. Published September 2, 2026.

Yuwei Guo, Ceyuan Yang, Ziyan Yang, Zhibei Ma, Zhijie Lin, Zhenheng Yang, Dahua Lin, and Lu Jiang. Long context tuning for video generation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 17281–17291, 2025.

Yoav HaCohen, Benny Brazowski, Nisan Chiprut, Yaki Bitterman, Andrew Kvochko, Avishai Berkowitz, Daniel Shalem, Daphna Lifschitz, Dudu Moshe, Eitan Porat, Eitan Richardson, Guy Shiran, Itay Chachy, Jonathan Chetboun, Michael Finkelson, Michael Kupchick, Nir Zabari, Nitzan Guetta, Noa Kotler, Ofir Bibi, Ori Gordon, Poriya Panet, Roi Benita, Shahar Armon, Victor Kulikov, Yaron Inger, Yonatan Shiftan, Zeev Melumian, and Zeev Farbman. Ltx-2: Efficient joint audio-visual foundation model. *arXiv preprint arXiv:2601.03233*, 2026. URL https://arxiv.org/abs/2601.03233.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*, 2021.

Panwen Hu, Jin Jiang, Jianqi Chen, Mingfei Han, Shengcai Liao, Xiaojun Chang, and Xiaodan Liang. Storyagent: Customized storytelling video generation via multi-agent collaboration. *arXiv preprint arXiv:2411.04925*, 2024.

Qixin Hu, Shuai Yang, Wei Huang, Song Han, and Yukang Chen. Longlive-rag: A general retrieval-augmented framework for long video generation. *arXiv preprint arXiv:2606.02553*, 2026.

Lianghua Huang, Wei Wang, Zhi-Fan Wu, Yupeng Shi, Huanzhang Dou, Chen Liang, Yutong Feng, Yu Liu, and Jingren Zhou. In-Context LoRA for diffusion transformers. *arXiv preprint arXiv:2410.23775*, 2024a.

Lingxuan Huang, Sizhe He, Hengji Zhou, Liqiang Nie, Lianghao Xia, and Chao Huang. Vimax: Agentic video generation. *arXiv preprint arXiv:2606.07649*, 2026.

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

Zhengyang Liang, Daoan Zhang, Huichi Zhou, Rui Huang, Bobo Li, Yuechen Zhang, Shengqiong Wu, Xiaohan Wang, Jiebo Luo, Lizi Liao, et al. Univa: Universal video agent towards open-source next-generation video generalist. *arXiv preprint arXiv:2511.08521*, 2025.

Han Lin, Abhay Zala, Jaemin Cho, and Mohit Bansal. Videodirectorgpt: Consistent multi-scene video generation via llm-guided planning. *arXiv preprint arXiv:2309.15091*, 2023.

Akide Liu, Jinbo Xing, Chaojie Mao, Ye Li, Zeyu Zhang, Yefei He, Weijie Wang, Zihan Wang, Yu Liu, Gholamreza Haffari, et al. Reca: Multi-shot long video extrapolation via recursive context allocation. *arXiv preprint arXiv:2605.26525*, 2026.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

Yihao Meng, Hao Ouyang, Yue Yu, Qiuyu Wang, Wen Wang, Ka Leong Cheng, Hanlin Wang, Shuailei Ma, Yixuan Li, Cheng Chen, et al. Holocine: Holistic generation of cinematic multi-shot long video narratives. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 461–471, 2026.

MiniMax. MiniMax H3: An open model breaking the boundaries between tasks and modalities, Jul 2026. URL https://www.minimax.io/blog/minimax-h3.

OpenAI. Sora 2 system card. OpenAI System Card, 2025. URL https://deploymentsafety.openai.com/sora-2.

OpenAI. Introducing ChatGPT Images 2.5, September 2026. URL https://openai.com/index/introducing-chatgpt-images-2-5/. Published September 8, 2026. Accessed September 25, 2026.

Tianhao Qi, Jianlong Yuan, Wanquan Feng, Shancheng Fang, Jiawei Liu, SiYu Zhou, Qian He, Hongtao Xie, and Yongdong Zhang. Maskˆ 2dit: Dual mask-based diffusion transformer for multi-scene long video generation. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 18837–18846, 2025.

Qwen Team. Qwen3.7-Plus: Multimodal agent intelligence, May 2026. URL https://qwen.ai/blog?id=qwen3.7-plus.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. DeepSeekMath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024.

Haoyuan Shi, Yunxin Li, Xinyu Chen, Longyue Wang, Baotian Hu, and Min Zhang. AniMaker: Multi-agent animated storytelling with MCTS-driven clip generation. In *Proceedings of the SIGGRAPH Asia 2025 Conference Papers*, pp. 1–11, 2025.

Yale Song, Yiwen Song, Nick Losier, Nathan Hodson, Ye Jin, Rhyard Zhu, Yan Xu, Daniel Vlasic, Carina Claassen, Jasmine Leon, et al. Co-director: Agentic generative video storytelling. *arXiv preprint arXiv:2604.24842*, 2026.

Xiaoxuan Tang, Xinping Lei, Chaoran Zhu, Shiyun Chen, Ruibin Yuan, Yizhi Li, Changjae Oh, Ge Zhang, Wenhao Huang, Emmanouil Benetos, et al. Automv: An automatic multi-agent system for music video generation. *arXiv preprint arXiv:2512.12196*, 2025.

Team Seedance, De Chen, Liyang Chen, Xin Chen, Ying Chen, Zhuo Chen, Zhuowei Chen, Feng Cheng, Tianheng Cheng, Yufeng Cheng, et al. Seedance 2.0: Advancing video generation for world complexity. *arXiv preprint arXiv:2604.14148*, 2026. URL https://arxiv.org/abs/2604.14148.

Tencent Hunyuan Foundation Model Team. Hunyuanvideo 1.5 technical report. *arXiv preprint arXiv:2511.18870*, 2025. URL https://arxiv.org/abs/2511.18870.

Wan Team, 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. URL https://arxiv.org/abs/2503.20314.

Qian Wang, Ziqi Huang, Ruoxi Jia, Paul Debevec, and Ning Yu. Mavis: A multi-agent framework for long-sequence video storytelling. In *Proceedings of the 19th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 2273–2295, 2026.

Weiyun Wang, Zhangwei Gao, Lixin Gu, Hengjun Pu, Long Cui, Xingguang Wei, Zhaoyang Liu, Linglin Jing, Shenglong Ye, Jie Shao, et al. InternVL3.5: Advancing open-source multimodal models in versatility, reasoning, and efficiency. *arXiv preprint arXiv:2508.18265*, 2025.

Yi Wang, Yinan He, Yizhuo Li, Kunchang Li, Jiashuo Yu, Xin Ma, Xinhao Li, Guo Chen, Xinyuan Chen, Yaohui Wang, et al. InternVid: A large-scale video-text dataset for multimodal understanding and generation. In *International Conference on Learning Representations*, volume 2024, pp. 42055–42079, 2024.

Weijia Wu, Zeyu Zhu, and Mike Zheng Shou. Automated movie generation via multi-agent cot planning. *arXiv preprint arXiv:2503.07314*, 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.

Tianyidan Xie, Zhentao Huang, Mingjie Wang, Xin Huang, Jun Zhou, Minglun Gong, and Zili Yi. Cineagi: Character-consistent movie creation through llm-orchestrated multi-modal generation and cross-scene integration. *arXiv preprint arXiv:2604.23579*, 2026.

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

Sikuan Yan, Xiufeng Yang, Zuchao Huang, Ercong Nie, Zifeng Ding, Zonggen Li, Xiaowen Ma, Jinhe Bi, Kristian Kersting, Jeff Z Pan, et al. Memory-R1: Enhancing large language model agents to manage and utilize memories via reinforcement learning. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 12805–12825, 2026.

Shuai Yang, Wei Huang, Ruihang Chu, Yicheng Xiao, Yuyang Zhao, Xianbang Wang, Muyang Li, Enze Xie, Yingcong Chen, Yao Lu, et al. Longlive: Real-time interactive long video generation. *arXiv preprint arXiv:2509.22622*, 2025.

Jung Yi, Wooseok Jang, Paul Hyunbin Cho, Jisu Nam, Heeji Yoon, and Seungryong Kim. Deep forcing: Training-free long video generation with deep sink and participative compression. *arXiv preprint arXiv:2512.05081*, 2025.

Xinlei Yin, Xiulian Peng, Xiao Li, Zhiwei Xiong, and Yan Lu. Closed-loop triplet synergistic generation for long-form video. *arXiv preprint arXiv:2606.16184*, 2026.

Jiwen Yu, Jianhong Bai, Yiran Qin, Quande Liu, Xintao Wang, Pengfei Wan, Di Zhang, and Xihui Liu. Context as memory: Scene-consistent interactive long video generation with memory retrieval. In *Proceedings of the SIGGRAPH Asia 2025 Conference Papers*, pp. 1–11, 2025a.

Qiying Yu, Zheng Zhang, Ruofei Zhu, Yufeng Yuan, Xiaochen Zuo, Yu Yue, Weinan Dai, Tiantian Fan, Gaohong Liu, Lingjun Liu, Xin Liu, Haibin Lin, Zhiqi Lin, Bole Ma, Guangming Sheng, Yuxuan Tong, Chi Zhang, Mofan Zhang, Wang Zhang, Hang Zhu, Jinhua Zhu, Jiaze Chen, Jiangjie Chen, Chengyi Wang, Hongli Yu, Yuxuan Song, Xiangpeng Wei, Hao Zhou, Jingjing Liu, Wei-Ying Ma, Ya-Qin Zhang, Lin Yan, Mu Qiao, Yonghui Wu, and Mingxuan Wang. DAPO: An open-source LLM reinforcement learning system at scale. arXiv preprint arXiv:2503.14476, 2025b. URL https://arxiv.org/abs/2503.14476.

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. *arXiv preprint arXiv:2512.19539*, 2025.

Peixuan Zhang, Zijian Jia, Kaiqi Liu, Shuchen Weng, Si Li, and Boxin Shi. Stage: Storyboard-anchored generation for cinematic multi-shot narrative. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 659–669, 2026.

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

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. *arXiv preprint arXiv:2601.03655*, 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.

Shaobin Zhuang, Kunchang Li, Xinyuan Chen, Yaohui Wang, Ziwei Liu, Yu Qiao, and Yali Wang. Vlogger: Make your dream a vlog. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8806–8817, 2024.

A Method and Training Details

A.1 Decision Context Mining and Dataset Statistics

Training multi-shot video generation end-to-end introduces severe compounding errors and intractable credit assignment over long temporal horizons. To address this, we localize policy updates to *single-shot transitions* ($t \to t+1$). Decision contexts $x_t = (o_t, \mathcal{M}_{t-1}, \mathcal{R}_t, u_t)$ are mined directly from multi-shot narrative video generation trajectories orchestrated by Gemini 3.5 Flash (Google DeepMind, 2026a) across 28 training scripts and 4 development scripts (each comprising a 12-shot video).

To prevent cross-shot narrative contamination across splits, contexts are partitioned strictly at the script level. The training corpus encompasses two operational modes:

Material Transitions. Teacher mutations are causally linked to subsequent shot continuity, ensuring that newly introduced or modified states are actively utilized in subsequent generation.

Preservation Controls. Memories are actively maintained without mutation despite ongoing character actions or minor visual drift, preventing unnecessary memory updates.

Transitions exhibiting unutilized teacher mutations or erroneous operations are strictly filtered out, yielding clean, causally verified decision contexts.

Each 12-shot script provides 11 candidate adjacent-shot transitions for constructing both datasets, giving an initial pool of 308 training transitions and 44 development transitions. Writing contexts undergo additional eligibility checks and manual screening to retain verified teacher updates with traceable links to next-shot references, together with valid no-write preservation controls. This produces 196 training contexts and 31 development contexts for the writing policy. Retrieval is required even when no write is performed, so its dataset also retains contexts in which the memory-update gate skips writing. The target retrieval answers are also manually reviewed and corrected where necessary. After excluding two training contexts with incomplete teacher annotations, the retrieval dataset contains 306 training contexts and 44 development contexts, for a total of 350.

Here, "image-free" refers to the absence of image generation during policy-training rollouts. The writing policy still receives visual observations from previously generated clips, while candidate memory updates are evaluated in structured text-attribute space without synthesizing new reference images.

A.2 Reward Formulation and Reachability Alignment

Writing Realization Reward (Q_c^{write}). For the memory writing policy $\pi_\theta^{\text{write}}$, the realization success indicator $Q_c^{\text{write}} \in \{0, 1\}$ evaluates the quality of proposed memory mutations under category $c \in \mathcal{C} = \{\text{character}, \text{scene}, \text{object}\}$. It is decomposed into four multiplicative factors:

$$Q_c^{\text{write}} = E_c \cdot D_c \cdot U_c \cdot J_c, \tag{4}$$

where $E_c \in \{0, 1\}$ denotes complete deterministic execution requiring strict adherence to formatting constraints, legal target identifiers, and valid attribute schemas; $D_c \in \{0, 1\}$ enforces non-duplication against pre-existing descriptions in entity catalog $\mathcal{S}_e$; $U_c \in \{0, 1\}$ requires downstream reference utility, ensuring newly created states are selected by a frozen downstream retrieval policy during shot continuity evaluation; and $J_c \in \{0, 1\}$ represents semantic alignment determined by a frozen text judge (qwen3.7-plus) comparing student mutations against expert teacher targets.

Retrieval Target Reachability Alignment (Q_c^{ret}). For the retrieval policy π_ϕ^{ret}, teacher target reference sets $P_c^\star$ are initially defined in the teacher's memory bank and may not directly align with indices in the student's materialized memory bank $\widetilde{\mathcal{M}}_t$. Directly supervising against teacher indices would unfairly penalize the retriever for upstream writer discrepancies. We therefore perform offline *target reachability alignment* along three principles:

Pre-Existing States. Carry forward invariant indices across narrative trajectories and require exact index matching.

Newly Created States. Audited semantically via a frozen text judge (Qwen3.7-Plus) against teacher targets; if a student-created state is verified to be semantically equivalent to a teacher target state, it is remapped to the corresponding student index in $\widetilde{\mathcal{M}}_t$.

Unreachable Categories. When the upstream writer omits a necessary state or produces ambiguous matches, the affected category is masked from supervision during training.

Following reachability alignment, the online retrieval reward is deterministically evaluated as $Q_c^{\text{ret}} = \mathbf{1}[P_c = P_c^\star]$.

A.3 Token-Level Value-Span Credit and Policy Optimization

Token-Level Value-Span Credit Assignment. Standard policy gradient methods assign uniform reward credit across all generated tokens, which dilutes policy gradients with non-semantic formatting tokens (such as JSON keys, brackets, and colons). EvoMA losslessly restricts policy updates strictly to tokens overlapping semantic value spans $\mathcal{T}_i^u$ for each decoupled unit $u = (c, \ell)$:

Arbitration Tokens. For $\ell = \text{arb}$, advantages apply exclusively to tokens representing decision keywords (e.g., NOOP, WRITE, EMPTY, SELECT).

Realization Tokens. For $\ell = \text{real}$, advantages apply exclusively to scalar value tokens specifying target entity identifiers, base state indices, and attribute descriptions within the proposal payload.

Syntactic delimiters, schema keys, and structural punctuation are assigned zero policy advantage, focusing gradients entirely on decision-making semantics.

Unit-Specific Advantage Standardization and Policy Objective. For each context x, we sample a group of G candidate completions $\{y_i\}_{i=1}^G$ from π_{old}. Across each decoupled unit $u = (c, \ell)$, the unit advantage $A_{x,i}^u$ is standardized independently across the subset of valid completions $\mathcal{I}_x^u$:

$$A_{x,i}^u = \begin{cases} \dfrac{r_{x,i}^u - \mu_x^u}{\sigma_x^u + \epsilon_{\text{num}}}, & \text{if } i \in \mathcal{I}_x^u,\ |\mathcal{I}_x^u| \geq 2,\ \sigma_x^u > 0, \\ 0, & \text{otherwise,} \end{cases} \tag{5}$$

where μ_x^u and σ_x^u denote the sample mean and standard deviation of rewards $\{r_{x,i}^u\}_{i \in \mathcal{I}_x^u}$, and $\epsilon_{\text{num}} = 10^{-6}$ ensures numerical stability. The per-token importance sampling ratio and clipped surrogate advantage are defined as:

$$\rho_{i,k}(\vartheta) = \frac{\pi_\vartheta(y_{i,k} \mid x, y_{i,<k})}{\pi_{\text{old}}(y_{i,k} \mid x, y_{i,<k})}, \qquad f(\rho, A) = \min\big(\rho A,\ \text{clip}(\rho, 1-\varepsilon, 1+\varepsilon)A\big), \tag{6}$$

where $\varepsilon = 0.2$ denotes the clipping threshold. The overall policy objective $\mathcal{J}(\vartheta)$ over trainable parameters $\vartheta \in \{\theta, \phi\}$ is then formulated as:

$$\mathcal{J}(\vartheta) = \mathbb{E}_{x \sim \mathcal{D},\, \{y_i\} \sim \pi_{\text{old}}} \left[\frac{1}{G} \sum_{i=1}^{G} \sum_u \frac{\omega_i^u}{\max(1, |\mathcal{T}_i^u|)} \sum_{k \in \mathcal{T}_i^u} f\big(\rho_{i,k}(\vartheta), A_{x,i}^u\big) - \beta\, \mathbb{D}_{\text{KL}}\big[\pi_\vartheta \parallel \pi_0\big] \right], \tag{7}$$

where ω_i^u balances positive action and state preservation across categories (detailed below), and $\beta = 0.01$ regularizes policy divergence from the raw base model π_0.

Category Objective Mass and Loss Weighting. Let $\mathcal{C}_+$ denote the set of supervised entity categories where the teacher target requires active mutation or selection (WRITE or SELECT), and let

$\mathcal{C}_0$ denote categories where the teacher target dictates preservation (NOOP or EMPTY). Across the ordered categories $(\text{character}, \text{scene}, \text{object})$, category priority vectors are defined as:

$$\boldsymbol{a}^+ = (1.0,\ 2.0,\ 1.5), \qquad \boldsymbol{a}^0 = (2.0,\ 1.0,\ 1.5). \tag{8}$$

The preservation scale factor λ_0 balances gradient updates between positive actions and state preservation:

$$\lambda_0 = \begin{cases} 0.3, & \mathcal{C}_+ \neq \emptyset, \\ 1.0, & \mathcal{C}_+ = \emptyset. \end{cases} \tag{9}$$

The objective mass m_c for category c is computed by normalizing priority weights within each active target set:

$$m_c = \begin{cases} \dfrac{a_c^+}{\sum_{d \in \mathcal{C}_+} a_d^+}, & c \in \mathcal{C}_+, \\ \dfrac{\lambda_0 a_c^0}{\sum_{d \in \mathcal{C}_0} a_d^0}, & c \in \mathcal{C}_0, \\ 0, & \text{masked categories.} \end{cases} \tag{10}$$

For each sampled completion y_i, category mass m_c is partitioned across arbitration unit g and realization unit p according to the sampled decision $g_{i,c}$:

$$(\omega_{i,c,g},\ \omega_{i,c,p}) = \begin{cases} (m_c/2,\ m_c/2), & g_{i,c} \in \{\text{WRITE},\ \text{SELECT}\}, \\ (m_c,\ 0), & g_{i,c} \in \{\text{NOOP},\ \text{EMPTY}\}. \end{cases} \tag{11}$$

These weights $(\omega_{i,c,g}, \omega_{i,c,p})$ act as post-advantage loss multipliers in the policy objective $\mathcal{J}(\vartheta)$, guaranteeing that category and decision units contribute balanced learning signals without mutual interference.

Dynamic Group Sampling. Because different entity categories converge at different rates during RL training, uniformly successful or failed prompts frequently produce zero reward variance, leading to wasted gradient updates. To maintain informative learning signals throughout training, we employ dynamic group sampling (Yu et al., 2025b). A sampled prompt group x of G completions is accepted into the active batch only when at least one valid unit exhibits non-zero reward variance:

$$\text{Accept}(x) = \mathbf{1}\left[\exists u :\ |\mathcal{I}_x^u| \geq 2 \ \wedge\ \sigma_x^u > 0\right]. \tag{12}$$

Candidate groups are resampled under a bounded budget until a batch of $B = 16$ informative groups is assembled.

B Additional Experimental Implementation Details

We instantiate both the memory writing policy $\pi_\theta^{\text{write}}$ and state retrieval policy π_ϕ^{ret} with Qwen3-VL-8B-Instruct as the shared base multimodal backbone. Both policies are trained with separate LoRA adapters (rank $r = 32$, $\alpha = 32$, dropout $= 0$) attached to language linear modules, while the base backbone weights remain frozen. Both policies are trained on $4\times$ NVIDIA H800 GPUs using bfloat16 mixed precision and optimized via AdamW (Loshchilov & Hutter, 2017) under GRPO with two-level credit assignment. We employ Qwen3.7-Plus as the frozen semantic text evaluator for evaluating candidate state realization alignment and target reachability (Appendix A.2). Table 5 provides the complete training hyperparameter configurations.

The writer adapter is trained over 60 rollout steps (120 optimizer steps), and the checkpoint after 50 rollout steps is selected based on dev-set reward. For the retrieval policy, this writer checkpoint is frozen to materialize student memory banks offline for the 306 training and 44 development contexts described in Appendix A.1. Across these 350 contexts, materialization invokes the frozen writer once for 263 contexts and reuses the captured memory bank without a writer call for the remaining 87. Retriever training proceeds for 25 rollout steps (50 optimizer steps), selecting the checkpoint after 25 rollout steps based on development-set reward, full-reward rate, and schema validity.

Table 5: Hyperparameter configurations for memory writing and retrieval policy training.

Hyperparameter	Memory Writer ($\pi_\theta^{\text{write}}$)	State Retriever (π_ϕ^{ret})
Base Backbone	Qwen3-VL-8B-Instruct	Qwen3-VL-8B-Instruct
Adapter Type	LoRA (language linear layers)	LoRA (language linear layers)
LoRA Rank (r) / Alpha (α)	32 / 32	32 / 32
LoRA Dropout	0.0	0.0
Optimizer	AdamW	AdamW
Learning Rate	2×10^{-5}	2×10^{-5}
AdamW β_1, β_2 / Weight Decay	$(0.9, 0.95)$ / 0.0	$(0.9, 0.95)$ / 0.0
Accepted Groups / Samples per Group	16 / 8	16 / 8
Updates per Rollout Batch	2	2
GRPO Clipping (ε) / KL Weight (β)	0.2 / 0.01	0.2 / 0.01
Gradient Norm Clip	1.0	1.0
Sampling Temperature / Top-p	1.0 / 0.95	1.0 / 0.95
Max Response Tokens	1536	512
Rollout Steps / Optimizer Steps	60 / 120	25 / 50
Hardware	$4\times$ NVIDIA H800	$4\times$ NVIDIA H800
Deployed Checkpoint (Rollout Step)	50	25

Memory-Agent Inference. In the default inference configuration, the global planner, memory writer, verifier, and retriever use temperature $T = 0$. When a clip is processed for memory writing, four frames are sampled per second at offsets of 0.10, 0.40, 0.60, and 0.90 seconds and arranged into a 2×2 mosaic for each second. The writer receives these mosaics together with relevant memory references and textual context. For each proposed write, the verifier receives up to six of the most recent frames from the cited seconds, together with relevant historical state references. Verification is performed by the unadapted base model.

C Benchmark Construction and Case Inventories

C.1 EvoVideo-Long Suite Design

All 32 scripts in the EvoVideo-Long suite were generated using Gemini-3.8-Flash (Google DeepMind, 2026c). Across these cases, 8 stories contain 8 shots, 16 contain 12 shots, and 8 contain 16 shots, yielding 384 shots. Across the benchmark, 141 unique entities are tracked across 489 physical and semantic states. The initial visual reference bundle contains 245 validated images (3 views per character, 1 primary image per scene or object). Table 6 provides the case inventory.

C.2 EvoVideo-Interactive Suite and Streaming Edits

The EvoVideo-Interactive suite contains 20 stories with 12 shots each (240 shots per method) paired with 60 scripted user edits (Table 7). Across the interactive suite, 83 total entities and 134 states are tracked (accounting for entities and states introduced or modified mid-stream through user interventions). The initial visual bundle comprises 74 entities and 122 reference images, with 9 additional entities introduced mid-stream across subsequent intervention shots.

Table 6: Inventory of the 32 test cases in the EvoVideo-Long suite.

Case	Title	Shots	Entities	States
1	The Railway Switchman's Midnight Blizzard	8	4	11
2	The Copper Etcher's Acid Darkroom	8	4	12
3	The Canal Lock's Tandem Haul	8	5	17
4	The Windmaster of the Cliff Weather Station	8	3	10
5	The Kiln Masters at the Firing Chamber	8	4	10
6	The Core Sampler in the Geological Trench	8	3	13
7	The Quenching Partners at the Blacksmith Forge	8	6	17
8	The Escapement Tuner atop the Clock Tower	8	4	13
9	The Leaking Janitor of the Neon Aquarium	12	3	13
10	The Crack Archivist on the Bone Bridge	12	3	14
11	The Wind-Kite Conservators of the Clock Tower	12	6	14
12	The Mud Rescue of the Rainforest Radio	12	4	16
13	The Backlit Magician of the Hourglass Courtyard	12	3	11
14	The Route Keepers of the Floating Archive	12	7	14
15	The Afterimage Tailor Duel Under the Moonlight	12	4	13
16	The Raging Tides of the Star Ferry Lantern Race	12	5	17
17	The Ice Sculptor in the Salt-Brine Cold Storage	12	4	18
18	Dual Watch at the Forest Fire Lookout	12	5	15
19	The Boiler Mechanic in the Locomotive Pit	12	3	13
20	Reassembling Sherds in the Archaeological Tent	12	6	18
21	Emergency Descent of the Deep-Mine Hoist Cage	12	6	22
22	The Glassblower's Ribbed Lantern Globe	12	3	15
23	Ice Core Drilling at the Glacier Station	12	5	14
24	The Letterpress Master's Multi-Color Run	12	6	15
25	The Cableway Mechanic at the Chasm Station	16	4	19
26	Water Quality Surveyors in the Shallow Mangrove	16	5	24
27	The Flying-Wing Tester in the Vintage Wind Tunnel	16	3	15
28	The Navigation Trio on the Stormy Bridge	16	7	21
29	The Peat Carbon Analyst at the Wetland Station	16	3	13
30	Valve Inspectors at the Geothermal Wellhead	16	5	20
31	The Cellar Masters and the Oak Tun	16	5	19
32	The Bow De-Icer on the Arctic Icebreaker	16	3	13

Table 7: Inventory of the 20 test cases and intervention schedules in EvoVideo-Interactive.

Case	Title	Entities	States	Edits	Intervention Shots (0-based)
1	Midnight at the Bakery	4	6	2	3, 7
2	Cable-Car Repairs Above the Valley	4	6	2	3, 7
3	Duet After the Rain	7	9	2	3, 7
4	Measuring the Desert Find	4	6	2	3, 7
5	Dawn at the Fish Market	4	5	2	3, 7
6	Three Changes at the Glassworks	4	6	3	2, 5, 8
7	Night Train Through Changing Scenery	4	7	3	2, 5, 8
8	Closing Time at the Museum	5	8	3	2, 5, 8
9	Testing the Coastal Prototype	4	6	3	2, 5, 8
10	The Garden Path After Rain	3	6	3	2, 5, 8
11	Rehearsal on Ice	4	6	3	2, 5, 8
12	The Tea House Above the Snowline	5	8	3	2, 5, 8
13	The Lost-and-Found Inventory	3	6	3	2, 5, 8
14	Changes on the Open-Air Stage	5	7	3	2, 5, 8
15	Maintenance in the Aquarium Tunnel	4	6	3	2, 5, 8
16	Midnight at the Neon Laundromat	4	7	4	2, 4, 6, 8
17	Waiting at the Rural Bus Stop	4	8	4	2, 4, 6, 8
18	A Changing Sky Above the Forge	4	7	4	2, 4, 6, 8
19	Tending the Lunar Greenhouse	3	7	4	2, 4, 6, 8
20	Rescue in the Flooded Library	4	7	4	2, 4, 6, 8

D Evaluation Protocols and Metrics

D.1 Evaluation Overview and Shared Scoring Rules

We evaluate generated shot-initial frames and the conditioning references actually selected for generation. The former measures identity consistency and state alignment in the generated output, while the latter diagnoses the suitability of the reference inputs. Both evaluations report identity (I) and state (S) scores, but use different identity anchors. For interactive generation, we evaluate identity consistency alongside immediate compliance with user edits and their persistence across subsequent shot openings.

Evaluation input	Identity comparison	State comparison
Generated shot-initial frame	Same-method, earliest clear frame of the entity	Expected state at the current shot opening
Selected reference image	Canonical initial reference of the entity	State required at the conditioned shot opening
Interactive shot-initial frame	Same-method, earliest clear frame within the entity's lifetime	Active, applicable user-edit requirements

Evaluations use the frozen Qwen3.7-Plus judge with temperature $T = 0$. Each scoring request concerns one target entity or edit target on one axis. A *match* receives 1; a conflict or unobservable required property receives 0. State matching requires visible support for all listed applicable requirements. Applicability is determined by the evaluation annotations: an annotated N/A item is excluded, whereas missing or ambiguous visual evidence for an applicable item does not receive an exemption. API failures remain pending and are not converted into zero scores. The scoring criteria are shared; the input construction and applicability rules for each evaluation are specified below.

D.2 Shot-Initial Frame Evaluation

Frame Extraction and Identity Anchors. We evaluate the first decoded frame of each generated shot, rather than its conditioning image or a later selected frame. For identity consistency, the current frame F is compared with the target entity's earliest clearly identifiable shot-initial frame A within the same method's generated story. Anchors are visually selected before scoring and are not changed in response to judge scores. The anchor shot and preceding shots are excluded from that entity's identity evaluation; identity is also excluded where no clear anchor exists or the original identity annotation is not applicable. These exclusions do not change the state-evaluation scope.

Identity and State Judgments. Identity scoring asks whether the target remains the same person, place, or object, allowing the annotated state changes. State scoring compares F with the annotated state expected at the current shot opening. Its scoring request contains the current image and expected state, without the anchor image or a sequence of past-state images. Consequently, identity consistency measures preservation relative to the method's own generated entity, whereas state alignment measures agreement with the script. The six category-specific scoring prompts are provided in Appendix J.

Presence and Target Localization. The identity and state evaluations share a target-presence result. In scenes requiring disambiguation, fixed entity identifiers, an entity roster, and textual location hints with their supporting reasons identify the target. A presence conflict sets applicable identity and state scores to zero. Otherwise, the corresponding axis judge evaluates the available evidence. State requests include only current-frame localization information; identity requests may include both anchor and current-frame localization. These hints identify the subject and do not replace visual evidence of its state.

D.3 Conditioning Reference Evaluation

Evaluation Inputs and Targets. We evaluate the actual entity reference images supplied to the image generator for each shot. References are assigned to benchmark entities using registered identities or audited provenance mappings; auxiliary assets that do not represent a benchmark entity

are excluded. Reference identity compares the selected image C with the entity's canonical initial reference A_0 from the benchmark bundle, using the front view for characters and the primary view for scenes and objects. Reference state scoring receives C and the annotated state required at the conditioned shot opening, without A_0 as a state target. Thus, a changed state may remain identity-consistent while differing from its initial appearance.

Multiple Views and Missing References. For a character state version, we retain the front view when it was actually selected; otherwise, we retain the actually selected alternative views. We then remove pixel-identical references within each entity–shot item. The remaining reference images receive equal weight, as formalized in Appendix D.4. This is image-level averaging after view selection and deduplication, rather than necessarily equal weighting of state versions. An applicable entity–shot item with no attributable reference receives zero; state items without applicable annotated requirements remain N/A. Reference scores therefore reflect both reference availability and the identity and state alignment of the selected images.

D.4 Hierarchical Aggregation and ISA

Let $x_{vtec}^{X,d} \in [0,1]$ denote the score for story v, shot t, entity e, category $c \in \mathcal{C} = \{\text{character}, \text{scene}, \text{object}\}$, axis $X \in \{I, S\}$, and evaluation domain $d \in \{\text{frame}, \text{ref}\}$. An applicable frame score is binary. For references, let $\mathcal{R}_{vtec}$ contain the images retained after view selection and pixel deduplication, and let $z_r^X \in \{0,1\}$ be the score for image r. The entity–shot reference score is

$$x_{vtec}^{X,\text{ref}} = \begin{cases} \dfrac{1}{|\mathcal{R}_{vtec}|} \displaystyle\sum_{r \in \mathcal{R}_{vtec}} z_r^X, & |\mathcal{R}_{vtec}| > 0, \\ 0, & |\mathcal{R}_{vtec}| = 0, \end{cases} \tag{13}$$

for applicable items only. This score may be fractional even though individual image judgments are binary.

For each domain and axis separately, let $\mathcal{E}_{vtc}^{X,d}$ contain applicable entities, $\mathcal{T}_{vc}^{X,d}$ contain shots with at least one applicable entity, and $\mathcal{V}_c^{X,d}$ contain stories with applicable shots. We compute

$$X_{vc}^d = \frac{1}{|\mathcal{T}_{vc}^{X,d}|} \sum_{t \in \mathcal{T}_{vc}^{X,d}} \frac{1}{|\mathcal{E}_{vtc}^{X,d}|} \sum_{e \in \mathcal{E}_{vtc}^{X,d}} x_{vtec}^{X,d}, \qquad X_c^d = \frac{1}{|\mathcal{V}_c^{X,d}|} \sum_{v \in \mathcal{V}_c^{X,d}} X_{vc}^d. \tag{14}$$

This hierarchy gives equal weight to applicable entities within a shot, applicable shots within a story, and stories within a category. It is not a pooled average over all entity–shot items. The category-level and overall *Identity–State Alignment* (ISA) scores are

$$\text{ISA}_c^d = \frac{I_c^d + S_c^d}{2}, \qquad \text{ISA}^d = \frac{1}{3} \sum_{c \in \mathcal{C}} \text{ISA}_c^d. \tag{15}$$

We report the two domains separately. ISA is the arithmetic mean of identity and state scores, not the fraction of items for which both judgments are simultaneously correct. The applicable sets for I and S can differ, including because of identity-anchor exclusions.

D.5 Interactive Identity, Edit Compliance, and Persistence

Evaluation Scope and Boundary Frames. Interactive evaluation measures identity consistency across benchmark entities, including entities introduced or replaced by user edits, and state alignment with active user-edit requirements. Identity is evaluated on script-applicable shot openings within each entity's lifetime; state evaluation is restricted to edit targets and does not separately score unedited background entities. Generation-time visibility and edit-delivery rules are specified in Appendix F.

For separately generated clips, we use decoded frame 0. For continuous LongLive outputs, we extract the first decoded frame at each audited prompt-switch boundary. The saved latent-to-output mapping places a boundary at latent index $L > 0$ at output-frame index $4L - 3$ (both zero-based). Frames are decoded sequentially to avoid approximate keyframe seeking. For state scoring, the judge receives the actual boundary frame, target-localization text, and applicable edit requirements, without memory-bank or reference images.

Identity Anchors and Entity Lifetimes. Interactive identity evaluation reuses the frozen category-specific identity prompts in Appendix J. Each entity is anchored to its earliest clearly identifiable shot-initial frame within its lifetime in the same method's output. Added entities enter at their effective edit boundary, and removed entities leave the identity-evaluation scope. A physical replacement ends the old entity's lifetime and starts a new identity track; clothing, hair, weather, lighting, and water-level changes preserve the existing track. Named groups are evaluated as groups rather than by matching individual members. Anchors are fixed before scoring, and their selection does not require successful realization of the requested edit. The anchor shot and preceding shots are excluded; entities with no clear anchor receive N/A for identity. On subsequent applicable shots, an absent target or unobservable identity receives zero. These identity exclusions do not alter state evaluation.

Identity Aggregation and ISA. We aggregate identity judgments using the entity–shot–story hierarchy in Appendix D.4, excluding N/A items and stories with no applicable identity items in the category. Mean I is the arithmetic mean across the three categories. Interactive ISA combines each category's identity score with its edited-state score as $\mathrm{ISA}_c = (I_c + S_c)/2$, and Mean ISA averages the three category scores. Identity covers a broader entity set than edited-state evaluation, so ISA summarizes the two axes over their respective applicable sets; it does not measure simultaneous identity and state success on a shared set of items.

Edited-State Requirements. All applicable requirements of a new edit must be visible at its onset boundary, with no grace period. Later shots are evaluated only for predeclared applicable requirements, and a superseded requirement ceases to be evaluated after its replacement takes effect. For removal edits, absence is successful only when the relevant inspection region is visible; an occluded or cropped region is unobservable.

Hierarchical State Score. For method m, let $y_i(m) \in \{0, 1\}$ be the score for an applicable edit-target item $i = (v, t, u)$ in category c. We apply the entity/target–shot–story hierarchy of Appendix D.4 to these edited-state items. Let $\mathcal{U}_{vtc}$ be the applicable edit targets, $\mathcal{T}_{vc}$ the applicable shots, and $\mathcal{V}_c$ the applicable stories. Each item has weight

$$w_i = \frac{1}{3\,|\mathcal{V}_c|\,|\mathcal{T}_{vc}|\,|\mathcal{U}_{vtc}|}, \qquad \mathrm{Mean_S}(m) = \sum_i w_i y_i(m), \qquad \sum_i w_i = 1. \tag{16}$$

The annotated state-evaluation sets are shared across methods.

Onset and Persistence under the Same Weights. Let $\mathcal{O}$ contain the 60 edit-onset items and $\mathcal{P}$ the 205 eligible post-onset items per method. These are evaluation items, not necessarily distinct image frames. Onset State Compliance and Temporal State Persistence preserve the weights of the overall state score:

$$S_{\mathrm{onset}}(m) = \frac{\sum_{i\in\mathcal{O}} w_i y_i(m)}{\sum_{i\in\mathcal{O}} w_i}, \qquad S_{\mathrm{persist}}(m) = \frac{\sum_{i\in\mathcal{P}} w_i y_i(m)}{\sum_{i\in\mathcal{P}} w_i}. \tag{17}$$

Writing $\alpha = \sum_{i\in\mathcal{O}} w_i$ gives the exact decomposition

$$\mathrm{Mean_S}(m) = \alpha S_{\mathrm{onset}}(m) + (1-\alpha) S_{\mathrm{persist}}(m). \tag{18}$$

For the frozen annotations, $\alpha \approx 0.2811495$ and $1 - \alpha \approx 0.7188505$. Post-onset evaluation includes edits that failed at onset, so persistence is not conditioned on initial success. It measures retention across subsequent shot openings, rather than continuous frame-by-frame persistence within each video segment.

E Detailed Quantitative Results

E.1 Six-Component Video Performance Breakdown

Table 8 provides the complete breakdown of all six component scores (Character I, Scene I, Object I, Character S, Scene S, Object S) for the 32-case long-video benchmark.

Table 8: Detailed six-component scores on the 32-case long-video benchmark.

Method	Character I	Scene I	Object I	Character S	Scene S	Object S
AniMaker	0.2946	0.3829	0.4377	0.2678	0.3888	0.3667
MovieAgent	**0.9692**	0.4469	0.4516	0.3844	0.3681	0.3550
ViMax	0.9048	0.7752	0.6525	0.3812	0.4499	0.4451
VideoMemory	0.8993	**0.9870**	0.6361	0.2907	0.5001	0.3642
EvoMA (Base)	0.8582	0.9844	0.8338	0.4912	0.7220	0.5733
EvoMA (GRPO)	0.8880	0.9800	**0.8806**	**0.5705**	**0.7596**	**0.6880**

E.2 Reference-Image Diagnostic Evaluation

Table 9 reports conditioning-reference results under the protocol in Appendix D.3, with the aggregation defined in Appendix D.4. The scores evaluate the actual entity reference images provided to the image generator.

Table 9: Reference-image six-component scores on the 32-case benchmark.

Method	Character I	Scene I	Object I	Character S	Scene S	Object S
VideoMemory	0.8824	0.9941	0.4042	0.2900	0.4710	0.2076
EvoMA (Base)	0.8931	0.9577	0.7791	0.5829	0.6725	0.5567
EvoMA (GRPO)	**0.9379**	**1.0000**	**0.8812**	**0.6523**	**0.7402**	**0.7295**

E.3 Performance by Story Length

Table 10 breaks down the EvoVideo-Long comparison by story length. EvoMA (GRPO) achieves the highest mean first-frame ISA in every group: 0.8220 for 8 shots, 0.8072 for 12 shots, and 0.7414 for 16 shots. Its 16-shot score is 0.0806 below its 8-shot score, indicating some sensitivity to longer narratives while retaining the lead over the other systems. Similarly, on conditioning reference images, EvoMA (GRPO) achieves leading consistency across all story lengths (0.8385, 0.8467, and 0.7622 mean ISA for 8, 12, and 16 shots, respectively).

E.4 Interactive Generation Results

Table 11 reports ISA and identity consistency on the 20-case EvoVideo-Interactive suite under the protocol in Appendix D.5. Identity is evaluated on the same generated shot-initial frames used for the state, onset, and persistence results in Table 3.

EvoMA (Base) achieves the highest Mean ISA (0.9226), followed by EvoMA (GRPO) (0.9168). Base also has higher Mean I (0.9319 vs. 0.9122), whereas GRPO has higher Mean S (0.9214 vs. 0.9133) and onset compliance (0.9238 vs. 0.8617); Base has higher persistence (0.9334 vs. 0.9205). These observations distinguish identity preservation from edit execution and retention. LongLive attains the highest Character I (0.9617), yet its Mean S is only 0.0937, illustrating that strong identity consistency alone does not establish successful user-edit realization. All scores concern shot-initial frames and do not measure consistency throughout every frame of a generated clip.

E.5 ST-Bench Evaluation Protocol and Setup

To evaluate out-of-distribution generalization beyond EvoVideoBench, we benchmark EvoMA on ST-Bench (Zhang et al., 2025), comprising 30 multi-shot narrative stories (300 video prompts).

Pipeline Configuration. We maintain the identical core pipeline as our main narrative experiment: Gemini-3.6-Flash for high-level global planning, GPT Image 2.5 Flare for keyframe reference rendering, Qwen3-VL-8B with our trained LoRA adapters for the memory writing policy ($\pi_\theta^{\text{write}}$,

Table 10: EvoVideo-Long performance by story length. ISA is measured on shot-initial frames and reference images ($[0, 1]$, higher is better). Scores are macro-averaged over cases within each group. Bold denotes the highest observed score in each column and group. Reference-image scores were evaluated only for VideoMemory and EvoMA.

	First Frame ISA				Reference Image ISA			
Method	Character	Scene	Object	Mean	Character	Scene	Object	Mean
8 shots (8 cases)								
AniMaker	0.2645	0.4405	0.5002	0.4017	N/A	N/A	N/A	N/A
MovieAgent	0.6724	0.4211	0.5013	0.5316	N/A	N/A	N/A	N/A
ViMax	0.6830	0.5848	0.6414	0.6364	N/A	N/A	N/A	N/A
VideoMemory	0.6367	0.7589	0.5993	0.6650	0.6719	0.7227	0.4128	0.6024
EvoMA (Base)	0.6820	0.8914	0.7225	0.7653	**0.8047**	0.8438	0.7448	0.7977
EvoMA (GRPO)	**0.7202**	**0.9018**	**0.8441**	**0.8220**	**0.8047**	**0.8594**	**0.8516**	**0.8385**
12 shots (16 cases)								
AniMaker	0.3454	0.3892	0.4345	0.3897	N/A	N/A	N/A	N/A
MovieAgent	0.7090	0.4220	0.4428	0.5246	N/A	N/A	N/A	N/A
ViMax	0.6652	0.6319	0.5536	0.6169	N/A	N/A	N/A	N/A
VideoMemory	0.6215	0.7302	0.4898	0.6138	0.6011	0.7786	0.2882	0.5560
EvoMA (Base)	0.7089	0.8302	0.7368	0.7586	0.7413	0.8255	0.6656	0.7441
EvoMA (GRPO)	**0.7624**	**0.8576**	**0.8014**	**0.8072**	**0.8142**	**0.8984**	**0.8275**	**0.8467**
16 shots (8 cases)								
AniMaker	0.1696	0.3244	0.2397	0.2446	N/A	N/A	N/A	N/A
MovieAgent	0.6166	0.3648	0.2261	0.4025	N/A	N/A	N/A	N/A
ViMax	0.5586	0.6014	0.4468	0.5356	N/A	N/A	N/A	N/A
VideoMemory	0.5004	0.7548	0.4215	0.5589	0.4707	0.6504	0.2344	0.4518
EvoMA (Base)	0.5989	0.8610	0.6181	0.6927	0.6647	0.7656	0.5957	0.6753
EvoMA (GRPO)	**0.6720**	**0.8622**	**0.6901**	**0.7414**	**0.7474**	**0.8242**	**0.7148**	**0.7622**

Table 11: Detailed interactive performance on EvoVideo-Interactive. We report Identity Consistency (I) and Integrated State-Aware Consistency (ISA) on shot-initial frames ($[0, 1]$, higher is better). Bold denotes the highest observed score in each column.

	Identity Consistency (I)				Integrated Consistency (ISA)			
Method	Character	Scene	Object	Mean	Character	Scene	Object	Mean
LongLive 1.0	**0.9617**	0.9667	0.7037	0.8773	0.5562	0.5042	0.3962	0.4855
AniMaker	0.6768	0.8444	0.5535	0.6916	0.5132	0.7488	0.5531	0.6050
MovieAgent	0.9000	0.2833	0.2043	0.4626	0.5470	0.4128	0.3092	0.4230
ViMax	0.8167	0.9333	0.3858	0.7120	0.5100	0.6714	0.4262	0.5358
EvoMA (Base)	0.9400	**0.9778**	**0.8780**	**0.9319**	**0.9482**	**0.9530**	0.8666	**0.9226**
EvoMA (GRPO)	0.9341	0.9389	0.8635	0.9122	0.9415	0.9403	**0.8686**	0.9168

rollout step 50) and state retrieval policy (π_ϕ^{ret}, rollout step 25), and base model verification for multimodal grounded verification. The video synthesis backend is adapted to Wan2.2 (832×464, 30 FPS).

Evaluation Metrics and Limitations. Following the official ST-Bench protocol (Zhang et al., 2025), evaluation uses LAION aesthetic predictor (Huang et al., 2024b) for visual aesthetic quality, and ViCLIP for prompt following and cross-shot consistency (Table 1). While EvoMA leads on four of five metrics (with unconstrained Wan2.2-T2V slightly higher on single-shot prompt following), these embedding-based similarity metrics primarily measure superficial visual resemblance and text matching. In stories where characters and environments remain largely static, high feature similarity rewards unchanging visual appearance rather than active memory dynamics. They fail to test whether an agent correctly detects mutations (e.g., costume changes, progressive damage, entity transfers), updates memory, and suppresses obsolete past states, underscoring the critical necessity of EvoVideoBench for rigorous dynamic state evaluation.

Table 12: Ablation results on EvoVideo-Long. We report ISA on shot-initial frames and reference images ($[0, 1]$, higher is better).

	First Frame ISA				Reference Image ISA			
Ablation Setting	Character	Scene	Object	Mean	Character	Scene	Object	Mean
Decoupled Policy Optimization (Section 3.3)								
EvoMA (Base)	0.6962	0.8591	0.7296	0.7616	0.8249	0.7839	0.7148	0.7745
+ Writing policy ($\pi_\theta^{\text{write}}$ only)	0.6783	**0.8944**	0.7512	0.7746	0.7522	0.8112	0.7207	0.7614
+ Retrieval policy (π_ϕ^{ret} only)	0.7044	0.8455	0.7714	0.7738	0.7867	0.8307	0.8353	0.8176
Full EvoMA (GRPO)	**0.7568**	0.8872	**0.8089**	**0.8176**	**0.8566**	**0.8451**	**0.8783**	**0.8600**
Training Paradigm								
EvoMA (SFT)	0.6945	0.8473	0.7505	0.7641	0.7901	0.7135	0.7682	0.7573
Multimodal Verification Layer (Section 3.2)								
Base w/o Verification	0.7100	0.8261	0.7144	0.7502	0.8379	0.7813	0.8125	0.8105
GRPO w/o Verification	0.7615	0.8239	0.7603	0.7819	0.8394	0.8008	0.8288	0.8230
Global Planner Generalization								
Global Qwen + Base	0.6744	0.8045	0.7091	0.7293	0.7331	0.7539	0.7383	0.7418
Global Qwen + GRPO	0.7541	0.8773	0.8085	0.8133	0.8108	0.8359	0.9342	0.8603

F Interactive Streaming Protocol and Runtime Audit

Let user edit e specify an effective 0-based arrival shot boundary $b(e) \in \{0, \ldots, T-1\}$. At shot index t, the set of newly arriving user edits is $\mathcal{U}_t^{\text{new}} = \{e : b(e) = t\}$, and the accumulated historical edit context is $\mathcal{U}_t^{\text{hist}} = \{e : b(e) \leq t\}$. Generation proceeds under strict online causality governed by five operational rules:

Causal Horizon and Future Masking. For shot t, generation observes only current instruction x_t, newly revealed edits $\mathcal{U}_t^{\text{new}}$, frozen initial reference assets $\mathcal{R}_0$ (per method interface), and causally produced historical outputs $y_{<t}$. Future narrative shots $x_{>t}$, future edits $\mathcal{U}_{>t}^{\text{new}}$, and future video frames remain strictly masked.

Strict Onset Boundary. The arrival boundary $b(e)$ is 0-based. An edit must take effect on the very first decoded frame of shot $b(e)$, without accommodation delays or warmup grace periods.

Boundary-Only Revelation. A user edit is revealed strictly at boundary $b(e)$ as an incremental directive. Adapters do not broadcast active edit ledgers or text summaries to subsequent shots.

History Immutability. Previously generated video clips $y_{<t}$ are strictly immutable; retroactive re-rendering, editing, or rollback is prohibited.

Accumulation and Superseding. Unrelated edits accumulate concurrently. When a subsequent edit targets the same entity attribute (e.g., Case 17 luggage, Case 18 weather, Case 19 tool, Case 20 water level, all arriving at shot index 8), the new requirement supersedes and terminates the prior requirement for that attribute.

Interactive identity, edit-compliance, and persistence evaluation follows Appendix D.5. Additional identity and ISA results are reported in Appendix E.4.

G Ablation Details and Memory Dynamics

The standardized ablation subset contains Cases 1, 7, 9, 10, 13, 21, 30, and 32 (96 total shots, 388 annotated entity–shot records).

G.1 Component Ablation Studies

Table 12 presents ISA evaluations on shot-initial frames alongside reference images.

Decoupled Policy Optimization. We assess the individual and joint contributions of the writing ($\pi_\theta^{\text{write}}$) and retrieval ($\pi_\phi^{\text{ret}}$) policies. As shown in Table 12, sequentially optimizing both policies

Table 13: Detailed six-component scores on the matched 8-case video ablation subset.

	Identity Consistency (I)			State Alignment (S)		
Ablation Setting	Character	Scene	Object	Character	Scene	Object
Decoupled Policy Optimization (Section 3.3)						
EvoMA (Base)	0.9094	**1.0000**	0.8485	0.4829	0.7181	0.6108
+ Writing policy ($\pi_\theta^{\text{write}}$ only)	0.9110	**1.0000**	0.8378	0.4456	**0.7889**	0.6646
+ Retrieval policy (π_ϕ^{ret} only)	0.9147	**1.0000**	0.8787	0.4940	0.6911	0.6640
Full EvoMA (GRPO)	**0.9435**	0.9896	**0.8936**	**0.5701**	0.7849	**0.7241**
Training Paradigm						
EvoMA (SFT)	0.9116	1.0000	0.8683	0.4773	0.6945	0.6327
Multimodal Verification Layer (Section 3.2)						
Base w/o Verification	0.8945	1.0000	0.8309	0.5256	0.6522	0.5980
GRPO w/o Verification	0.9332	1.0000	0.8204	0.5898	0.6478	0.7002
Global Planner Generalization						
Global Qwen + Base	0.9304	1.0000	0.8214	0.4184	0.6089	0.5967
Global Qwen + GRPO	0.9416	1.0000	0.9013	0.5666	0.7545	0.7157

substantially outperforms optimizing either policy in isolation (0.8176 vs. 0.7746 and 0.7738 ISA on shot-initial frames). Optimizing the writer alone drops reference image quality when the unoptimized retriever fails to fetch updated states, while optimizing the retriever alone cannot accommodate novel mutations.

Training Paradigm. We compare GRPO with two-level credit assignment against supervised fine-tuning (SFT) on teacher trajectories. EvoMA (SFT) yields negligible gains over the base model (0.7641 vs. 0.7616 ISA on shot-initial frames) because it over-fits to teacher demonstrations by making minimal memory updates, thereby missing crucial entity additions and state updates, which validates our decoupled reinforcement learning formulation.

Multimodal Verification Layer. Disabling verification allows spurious mutations to accumulate in memory (Table 15), consistently degrading downstream video consistency for both Base and GRPO (Table 12). While unverified Base admits unconstrained proposals that superficially elevate reference-image matching, this gain fails in video generation. For full EvoMA (GRPO), verification protects both reference conditioning and downstream video fidelity.

Global Planner Generalization. We test whether EvoMA's learned memory policies generalize across different narrative planning backbones by replacing Gemini-3.6-Flash with open-source Qwen3-VL-8B. GRPO achieves a consistent +0.0840 ISA gain on shot-initial frames, confirming that our memory policy optimization is robust to the choice of global LLM.

G.2 Detailed Six-Component Ablation Breakdown

Table 13 and Table 14 present the complete six-component breakdowns for shot-initial video frames and conditioning reference images, respectively. Across both evaluations, scene identity remains near ceiling across all configurations, whereas policy optimization predominantly improves dynamic state alignment (S). In particular, full EvoMA (GRPO) achieves marked gains in character state alignment (0.5701 vs. 0.4829 on video, 0.7613 vs. 0.7096 on references) and object state alignment (0.7241 vs. 0.6108 on video, 0.8164 vs. 0.6133 on references) over Base. Furthermore, optimizing either policy in isolation yields suboptimal state alignment on complementary axes (e.g., character state on video drops to 0.4456 under writer-only optimization). These results support sequentially optimizing both policies rather than optimizing either policy alone on the evaluated subset.

G.3 Memory Operation Dynamics and Redundancy Audit

Table 15 reports operational memory execution statistics across the matched 8-case subset (96 shots).

Table 14: Detailed six-component reference-image scores on the matched 8-case ablation subset.

	Identity Consistency (I)			State Alignment (S)		
Ablation Setting	Character	Scene	Object	Character	Scene	Object
Decoupled Policy Optimization (Section 3.3)						
EvoMA (Base)	0.9401	0.9583	0.8164	0.7096	0.6094	0.6133
+ Writing policy ($\pi_\theta^{\text{write}}$ only)	0.9253	0.9896	0.7500	0.5790	0.6328	0.6914
+ Retrieval policy (π_ϕ^{ret} only)	0.9484	1.0000	0.9049	0.6250	0.6615	0.7656
Full EvoMA (GRPO)	**0.9518**	**1.0000**	**0.9401**	**0.7613**	**0.6901**	**0.8164**
Training Paradigm						
EvoMA (SFT)	0.9557	0.9896	0.8242	0.6246	0.4375	0.7122
Multimodal Verification Layer (Section 3.2)						
Base w/o Verification	0.9336	0.9896	0.8581	0.7422	0.5729	0.7669
GRPO w/o Verification	0.9484	0.9896	0.8776	0.7305	0.6120	0.7799
Global Planner Generalization						
Global Qwen + Base	0.9332	0.9714	0.8008	0.5330	0.5365	0.6758
Global Qwen + GRPO	0.9327	1.0000	0.9596	0.6888	0.6719	0.9089

Table 15: Operational memory execution statistics across the matched 8-case subset (96 shots). Final denotes registered final entity-state versions; Proposed counts memory write proposals; Exact Dups counts identical within-entity descriptions; Verifier counts logical verification requests; Coverage denotes the fraction of the 388 required entity-shot pairs supplied with at least one attributable reference. Counts are pooled across cases and differ from macro-averaged ISA.

Ablation Setting	Final	Proposed	Exact Dups	Verifier	Coverage
Decoupled Policy Optimization (Section 3.3)					
EvoMA (Base)	95	116	0	69	86.1%
+ Writing policy ($\pi_\theta^{\text{write}}$ only)	102	87	0	82	84.8%
+ Retrieval policy (π_ϕ^{ret} only)	99	108	0	76	91.0%
Full EvoMA (GRPO)	109	87	0	80	**94.8%**
Training Paradigm					
EvoMA (SFT)	85	80	0	58	88.7%
Multimodal Verification Layer (Section 3.2)					
Base w/o Verification	128	100	18	0	89.2%
GRPO w/o Verification	111	79	8	0	89.9%
Global Planner Generalization					
Global Qwen + Base	108	133	0	91	88.1%
Global Qwen + GRPO	106	93	0	82	94.3%

As shown in Table 15, SFT retains substantially fewer final entity states (85 vs. 95 for Base and 109 for GRPO), reflecting conservative behavior that fails to capture entity mutations. Meanwhile, full EvoMA (GRPO) achieves the highest required-reference coverage (94.8% vs. 86.1% for Base and 89.9% for unverified GRPO). Crucially, disabling verification permits spurious redundant states to accumulate (18 and 8 exact duplicates for Base and GRPO, respectively). Incorporating our multimodal verification layer completely eliminates duplicate state registrations (0 across all verified settings) while safeguarding downstream reference quality.

G.4 Cross-Backbone Training Exploration

To evaluate whether our decoupled reinforcement learning formulation generalizes across multimodal model architectures and parameter scales, we train and evaluate EvoMA policies over two alternative vision-language backbones: compact **Qwen3-VL-4B** (Bai et al., 2025) and architecture-distinct **InternVL3.5-8B** (Wang et al., 2025). For each backbone family, we evaluate the original instruction base model against GRPO-optimized checkpoints for both memory writing and reference retrieval under the shared generation pipeline.

Table 16: Cross-backbone policy generalization on EvoVideo-Long (8-case matched subset). We evaluate original instruction-model base policies against GRPO-optimized checkpoints for both memory writing and retrieval ($[0, 1]$, higher is better).

	First Frame ISA				Reference Image ISA			
Ablation Setting	Character	Scene	Object	Mean	Character	Scene	Object	Mean
Qwen3-VL-4B Backbone								
Base	0.6429	0.8388	0.7341	0.7386	0.7461	0.7630	0.6888	0.7326
Full EvoMA (GRPO)	**0.7061**	**0.8549**	**0.7526**	**0.7712**	**0.8171**	**0.8451**	**0.7513**	**0.8045**
InternVL3.5-8B Backbone								
Base	0.6656	0.7568	0.6720	0.6981	0.7181	0.5885	0.4219	0.5762
Full EvoMA (GRPO)	**0.6991**	**0.8203**	**0.7604**	**0.7599**	**0.7806**	**0.8229**	**0.7318**	**0.7784**

As shown in Table 16, GRPO consistently elevates end-to-end generation quality across both model families. On Qwen3-VL-4B, GRPO improves mean shot-initial-frame ISA from 0.7386 to 0.7712 and reference-image ISA from 0.7326 to 0.8045. On InternVL3.5-8B, GRPO improves shot-initial-frame ISA from 0.6981 to 0.7599 and substantially boosts reference-image ISA from 0.5762 to 0.7784 (+0.2022). These paired gains demonstrate that our decoupled policy optimization framework transfers effectively across varying parameter scales and model architectures.

H User Study Protocol and Detailed Results

H.1 Viewing Interface and Rating Task

The interface defaults to side-by-side shot-initial frames, with synchronized shot navigation for A and B. Participants may switch to the full videos; the interface recommends video viewing but permits frame-only evaluation. Script-derived expected states are displayed below the frames, and participants can consult a state-change summary or the complete script. The guidance emphasizes whether a changed state persists until the script specifies a subsequent change, while also considering persistent entity identity. These annotations describe intended states, not automated judgments of the generated images.

For each pair, participants separately rate character, scene, object, and overall consistency. The category-specific questions concern both entity identity and script-consistent visual states. Overall consistency is an independent holistic judgment, not an average of the other three ratings. Each question permits four responses: A is better, B is better, comparable (tie), or unable to judge. Thus, the task is not a forced binary choice.

H.2 Win-Rate Definition

For method m and rating dimension d, let $W_{m,d}$, $L_{m,d}$, and $T_{m,d}$ denote wins, losses, and ties after mapping the anonymous A/B responses to method identities. We compute

$$\mathrm{WinRate}_{m,d} = \frac{W_{m,d}}{W_{m,d} + L_{m,d} + T_{m,d}} \times 100\%. \tag{19}$$

Ties remain in the denominator but do not count as wins; unable-to-judge responses are excluded from the denominator and reported separately. Each comparison contributes one outcome to each of its two methods in each dimension.

I System Prompts for Memory Writing, Verification, and Retrieval

I.1 Writing Policy

The system prompts below correspond to the streamlined pipeline used in our experiments. This implementation omits a separate writer reasoning-trace output: the writer directly returns structured memory-action proposals, which are verified against visual evidence and historical memory.

The writer first decides whether to keep or write memory for each category. Within a category selected for writing, it proposes entity-level additions or updates supported by the current clip. The writer predicts the target entity identifier, reusable state description, and evidence timestamps; the controller resolves the base-state index. The implementation token `KEEP` corresponds to the no-write decision denoted NOOP in the method formulation.

Writer system prompt

```
[Role]
You are the memory updater for a video agent. Use the current clip, next-shot
context, first-frame state selections, and existing memory to maintain visual
continuity for later clips.

[State selections]
current_state_selections contains the memory states planned for the first frame
of THIS clip. Every selection has exactly one type, entity idx, and state_idx.
Writes must describe canonical entity changes supported by this clip. For an
UPDATE, the controller deterministically resolves the existing base state from
these selections. Do not output an entity index or state index.
The retriever will select the memory states for the NEXT clip using the same
selection format.

[Memory]
- character: a recurring person's identity and reusable visible appearance.
- environment: a recurring place and its reusable visual condition.
- noteworthy: a recurring object or visual detail whose state matters later.
- static_description is stable entry identity, not a dynamic state.
- states contain text-only reusable state descriptions. Style is read-only in
  this profile.

[Available writes]
- add registers a genuinely new recurring entity.
- update appends a materially new reusable state to an existing entity.

[Fixed category slots]
- Output exactly three top-level slots in this order: character, environment,
  noteworthy. Never omit, rename, or add a category slot.
- Every slot must first make an explicit gate decision.
- "decision" and each write's "op" are different fields. Emit the canonical
  decision strings "KEEP" and "WRITE".
- KEEP means the current memory already covers this category, or no durable
  visible change is justified. KEEP requires an empty writes list.
- WRITE means this category needs one or more durable memory writes. WRITE
  requires a non-empty writes list.
- The enclosing slot determines the memory type. Do not output type inside an
  individual write. The controller injects it deterministically.
- Decide each slot independently. A clip may WRITE character and environment
  while KEEPING noteworthy, or use any other evidence-supported combination.
```

Writer system prompt (continued)

```
[Write policy]
- For a write whose op is "update", id must copy the exact string entry id from
  memory, for example character_Alex. Never output an entity idx, state_idx,
  base_state_idx, or target_state_idx.
- Write only durable visual state that can affect continuity in a later clip.
- Do not write one-time actions, poses, camera or framing changes, rendering
  errors, or minor visual drift.
- Use add for a genuinely new entity and update for a materially new reusable
  state.
- Every write must cite visible evidence from this clip with at least one
  evidence_seconds item. Evidence seconds are zero-based clip seconds.
- For a clip whose duration is D seconds, evidence_seconds may contain only
  integers from 0 through D-1. For example, a 5-second clip allows 0,1,2,3,4;
  never cite the end boundary 5.
- state_description must describe the reusable current visible state, not only
  the action that produced it. Do not rewrite or delete old entity states.
- Character, environment, and noteworthy writes are text-only. Do not output
  state features.
- Emit at most one write for the same id or name within a slot.
- next_shot is continuity context only. Never use it as visual evidence for a
  write unless the state is visibly present in the current clip.

[Output]
Return JSON only. The JSON below is only a format example, not the answer.
Omit write fields that do not apply, but never omit a category slot, decision,
or writes list.
{
  "character": {
    "decision": "WRITE",
    "writes": [
      {
        "op": "update",
        "id": "<exact existing entry id; update only>",
        "state_description": "<reusable visible state>",
        "evidence_seconds": [0, 2]
      }
    ]
  },
  "environment": {
    "decision": "KEEP",
    "writes": []
  },
  "noteworthy": {
    "decision": "KEEP",
    "writes": []
  }
}
When nothing should change, return all three slots with decision KEEP and an
empty writes list in each slot.
```

I.2 GROUNDED VERIFIER

The verifier processes one controller-resolved write at a time, using attached source frames and relevant memory context. Its output specifies a verdict and, when applicable, an executable memory action.

Verifier system prompt

```
[Role]
You are the memory verifier for a video agent. Verify one controller-resolved
write using the attached current-clip frames, relevant memory references, and
story context. Return APPROVE, MODIFY, or REJECT.

[Decision]
- APPROVE when the write is visibly supported, durable, useful for later
  continuity, correctly targeted, and executable as written.
- MODIFY when the evidence supports a memory write but its action, target,
  base state, or state text must be corrected.
- REJECT when evidence is missing or ambiguous, the state is transient, the
  target is invalid, or existing memory already covers the visible state.
- Use action_type=NONE when no registry write is needed.

[Rules]
- ADD and UPDATE require at least one attached source frame from this clip.
  Story text is context, not visual evidence.
- Store only durable reusable visual state. Reject one-time actions, poses,
  camera/framing changes, rendering errors, and minor drift.
- ADD creates a genuinely new recurring character, environment, or noteworthy
  entity. UPDATE appends a materially new state to an existing exact target_id
  and existing target_state_idx; it never rewrites an old state.
- For ADD, set target_id and target_state_idx to null and provide target_name.
  For UPDATE, keep an exact existing target_id and base target_state_idx.
- Keep the proposal's memory_type. The writer's fixed category slot is
  authoritative; MODIFY may correct ADD/UPDATE/NONE, target, base state, text,
  or source frame, but must not move a write to another memory category.
- proposed_text describes the reusable visible state, not only the action that
  produced it.
- For ADD or UPDATE, choose source_frame_for_state from the attached source
  filenames. Prefer the clearest frame showing the completed state.
- This profile is text-only and supports only character, environment, and
  noteworthy ADD/UPDATE writes. Do not output style, DELETE, state features,
  or fields not shown in the schema.

[Output]
Return JSON only. The shape below illustrates an UPDATE; use null for the ADD
target fields as required above.
{
  "verdict": "APPROVE | MODIFY | REJECT",
  "final_action": {
    "action_type": "ADD | UPDATE | NONE",
    "memory_type": "character | environment | noteworthy",
    "target_id": "<exact existing id or null>",
    "target_name": "<new entity name or null>",
    "target_state_idx": 0,
    "proposed_text": "<verified reusable visible state>",
    "source_frame_for_state": "<attached source filename or null>"
  },
  "reason": "<brief>"
}
If verdict is REJECT, final_action may be null.
```

I.3 RETRIEVAL POLICY

The retriever selects existing entity-state references for the next clip's first frame. For each memory category, it decides whether references are needed and, if so, returns the entity and state indices of the selected entries.

Retriever system prompt

```
[Role]
You select memory references for the video agent's NEXT clip only.

[Task]
Use the next-shot description and available memory to select the existing
character, environment, and noteworthy states needed at the first frame of
the next clip.

[Rules]
- Output exactly three top-level slots in this order: character, environment,
  noteworthy. Never omit, rename, or add a category slot.
- Each slot must make an explicit decision: SELECT or EMPTY.
- SELECT requires a non-empty selections list. EMPTY requires an empty list.
- Every selection contains only idx and state_idx. The enclosing slot supplies
  the memory type.
- Select only entries and state_idx values present in available_memory.
- Include only entities visible or materially needed in the next clip.
- A state marked created_in_current_shot may be selected when it continues
  into the next clip.
- Select at most one environment and at most one state for each entity.
- Do not invent entries, indices, states, or story details.

[Output]
Return JSON only. The JSON below is a format example, not the answer.
{
  "character": {
    "decision": "SELECT",
    "selections": [{"idx": 0, "state_idx": 1}]
  },
  "environment": {
    "decision": "EMPTY",
    "selections": []
  },
  "noteworthy": {
    "decision": "SELECT",
    "selections": [{"idx": 2, "state_idx": 0}]
  }
}
When no reference is needed for a category, use EMPTY for that category only.
```

J SHOT-INITIAL FRAME EVALUATION PROMPTS

We provide the system prompts used for automated identity consistency (I) and state alignment (S) evaluation (Appendix D.2), where T denotes the target entity, A its identity anchor frame, and F the evaluated shot-initial frame.

J.1 CHARACTER EVALUATION

Character Identity Consistency (I)

```
[Task]
Is T the same person or character? Use visible facial geometry and distinctive
    facial features. Hair color/style, facial hair, makeup, wrinkles or
    apparent age, clothing and accessories alone do not establish a different
    person. For a character defined by a permanent mask, compare its
    distinctive design; ordinary protective gear is not a face substitute. If
    identifying features are hidden, return unobservable.
A is T's fixed earliest clear shot-first-frame in this generated video. F is a
    later shot-first-frame from the same video. Judge internal identity
    consistency, not agreement with an external design.
[Rules]
Find only T in each full image using the supplied target hints. Hints locate
    the target; they are not appearance requirements. In group scenes, do not
    transfer the name to a companion because clothing, position or relative
    height changed. If target assignment is ambiguous, return unobservable.
Allow viewpoint, pose, lighting and state changes. Damage, repairs, clothing
    and listed STATE_CHANGES alone do not establish a different identity. Do
    not require F to copy A's state or framing. Use visible evidence, not the
    name or story as proof of sameness.
Return match when visible identity features support the same entity. Return
    mismatch only for clear identity-bearing differences that are not
    explained by state, viewpoint, expression or lighting; name those
    differences in the reason. Otherwise return unobservable. Small, blurred,
    occluded or rear views do not by themselves prove mismatch. Do not invent
    hidden details.
[Output]
JSON only: {"verdict":"match|mismatch|unobservable","reason":"Brief decisive
    visible evidence or limitation."}
```

Character State Alignment (S)

```
[Task]
Does target T in first frame F have the full EXPECTED opening state?
[State]
State is reusable visible appearance: clothing and footwear, worn accessories,
    hairstyle, stains/wetness, injuries, bandages and bodily transformations.
    Removal or replacement of something worn changes state. Holding an
    independent prop, facial expression, action and pose alone do not. Judge
    only this person, not a nearby companion.
[Rules]
Previously established conditions persist until explicitly changed; judge the
    supplied opening state, not the action that will happen next. Do not
    require motion direction or flicker frequency from one still image.
EXPECTED already accounts for the scripted opening composition. Do not
    evaluate later actions or add requirements. Inspect the relevant visible
    regions before deciding; expected text is not evidence that a condition is
    present.
Use anatomical left/right for body parts, accounting for orientation. Judge
    the target itself, not nearby objects, shadows or reflections. Do not
```

Character State Alignment (S) (continued)

```
    infer hidden contents from surface shine or infer missing details from the
    story.
Respect specified degree, amount, and spatial relations. Read the description
    together: repairs can retain specified damage, replacement differs from
    addition, and optional details are not mandatory. Ignore identity quality,
    pose, framing style and minor rendering differences.
Return conflict if a requirement is visibly contradicted; otherwise match only
    when the full required state has visible support; otherwise unobservable.
    Cropping/occlusion of a required detail is insufficient evidence, not
    automatic conflict or permission to omit it.
[Output]
JSON only: {"observation":"Brief relevant visible
    evidence.","verdict":"match|conflict|unobservable","reason":"Decisive
    agreement, contradiction or visibility limit."}
```

J.2 Scene Evaluation

Scene Identity Consistency (I)

```
[Task]
Is T the same physical place? Compare spatial layout, fixed structures and
    landmarks. A shared scene category, vegetation or color palette alone is
    insufficient.
A is T's fixed earliest clear shot-first-frame in this generated video. F is a
    later shot-first-frame from the same video. Judge internal identity
    consistency, not agreement with an external design.
[Rules]
Find only T in each full image using the supplied target hints. Hints locate
    the target; they are not appearance requirements. In group scenes, do not
    transfer the name to a companion because clothing, position or relative
    height changed. If target assignment is ambiguous, return unobservable.
Allow viewpoint, pose, lighting and state changes. Damage, repairs, clothing
    and listed STATE_CHANGES alone do not establish a different identity. Do
    not require F to copy A's state or framing. Use visible evidence, not the
    name or story as proof of sameness.
Return match when visible identity features support the same entity. Return
    mismatch only for clear identity-bearing differences that are not
    explained by state, viewpoint, expression or lighting; name those
    differences in the reason. Otherwise return unobservable. Small, blurred,
    occluded or rear views do not by themselves prove mismatch. Do not invent
    hidden details.
[Output]
JSON only: {"verdict":"match|mismatch|unobservable","reason":"Brief decisive
    visible evidence or limitation."}
```

Scene State Alignment (S)

```
[Task]
Does target T in first frame F have the full EXPECTED opening state?
[State]
State is the reusable condition of the same place: ambient light/time,
    weather, ground or water condition, fixed layout, landmarks and structural
    damage or repairs. A changed viewpoint is not a changed layout. A movable
    object belongs to its own entity, unless EXPECTED explicitly describes its
    relation to the place.
[Rules]
```

Scene State Alignment (S) (continued)

```
Previously established conditions persist until explicitly changed; judge the
    supplied opening state, not the action that will happen next. Do not
    require motion direction or flicker frequency from one still image.
EXPECTED already accounts for the scripted opening composition. Do not
    evaluate later actions or add requirements. Inspect the relevant visible
    regions before deciding; expected text is not evidence that a condition is
    present.
Use anatomical left/right for body parts, accounting for orientation. Judge
    the target itself, not nearby objects, shadows or reflections. Do not
    infer hidden contents from surface shine or infer missing details from the
    story.
Respect specified degree, amount, and spatial relations. Read the description
    together: repairs can retain specified damage, replacement differs from
    addition, and optional details are not mandatory. Ignore identity quality,
    pose, framing style and minor rendering differences.
Return conflict if a requirement is visibly contradicted; otherwise match only
    when the full required state has visible support; otherwise unobservable.
    Cropping/occlusion of a required detail is insufficient evidence, not
    automatic conflict or permission to omit it.
[Output]
JSON only: {"observation":"Brief relevant visible
    evidence.","verdict":"match|conflict|unobservable","reason":"Decisive
    agreement, contradiction or visibility limit."}
```

J.3 Object Evaluation

Object Identity Consistency (I)

```
[Task]
Is T the same object or named object group? Compare distinctive design,
    structure, parts and markings. A shared object category or color alone is
    insufficient.
A is T's fixed earliest clear shot-first-frame in this generated video. F is a
    later shot-first-frame from the same video. Judge internal identity
    consistency, not agreement with an external design.
[Rules]
Find only T in each full image using the supplied target hints. Hints locate
    the target; they are not appearance requirements. In group scenes, do not
    transfer the name to a companion because clothing, position or relative
    height changed. If target assignment is ambiguous, return unobservable.
Allow viewpoint, pose, lighting and state changes. Damage, repairs, clothing
    and listed STATE_CHANGES alone do not establish a different identity. Do
    not require F to copy A's state or framing. Use visible evidence, not the
    name or story as proof of sameness.
Return match when visible identity features support the same entity. Return
    mismatch only for clear identity-bearing differences that are not
    explained by state, viewpoint, expression or lighting; name those
    differences in the reason. Otherwise return unobservable. Small, blurred,
    occluded or rear views do not by themselves prove mismatch. Do not invent
    hidden details.
[Output]
JSON only: {"verdict":"match|mismatch|unobservable","reason":"Brief decisive
    visible evidence or limitation."}
```

Object State Alignment (S)

```
[Task]
Does target T in first frame F have the full EXPECTED opening state?
[State]
State is a recurring object's visible condition: material/color, shape, parts,
    contents, open/closed status, light emission, damage, repairs and
    attachment or placement when specified. A removed part differs from an
    added part; a repair may leave damage. Changed appearance does not itself
    imply changed identity.
[Rules]
Previously established conditions persist until explicitly changed; judge the
    supplied opening state, not the action that will happen next. Do not
    require motion direction or flicker frequency from one still image.
EXPECTED already accounts for the scripted opening composition. Do not
    evaluate later actions or add requirements. Inspect the relevant visible
    regions before deciding; expected text is not evidence that a condition is
    present.
Use anatomical left/right for body parts, accounting for orientation. Judge
    the target itself, not nearby objects, shadows or reflections. Do not
    infer hidden contents from surface shine or infer missing details from the
    story.
Respect specified degree, amount, and spatial relations. Read the description
    together: repairs can retain specified damage, replacement differs from
    addition, and optional details are not mandatory. Ignore identity quality,
    pose, framing style and minor rendering differences.
Return conflict if a requirement is visibly contradicted; otherwise match only
    when the full required state has visible support; otherwise unobservable.
    Cropping/occlusion of a required detail is insufficient evidence, not
    automatic conflict or permission to omit it.
[Output]
JSON only: {"observation":"Brief relevant visible
    evidence.","verdict":"match|conflict|unobservable","reason":"Decisive
    agreement, contradiction or visibility limit."}
```