VAEM: GIVE VIDEO AVATARS AN EVOLVING MIND

Anonymous authors
Paper under double-blind review

ABSTRACT

Recent efforts have explored endowing video avatars with active intelligence to generate multi-step video sequences toward high-level goals. However, existing methods fail to consolidate past failures into persistent knowledge, frequently repeating similar errors in planning and execution. To address this, we present VAEM (Video Avatar Evolving Mind), a framework that distills video-generation histories into a persistent planner context comprising behavioral principles in an evolving system prompt and reusable skills in a structured library. VAEM operates via a Plan-Generate-Critique-Update (PGCU) cycle, orchestrating a scene-aware planner, a frozen video generator, specialized vision-language critics, and an update agent. During context evolution, the critics localize semantic failures, enabling the update agent to convert feedback into actionable rules and skills for subsequent tasks. At inference, the accumulated context is frozen and reused directly for planning without online updates. Extensive experiments on half-body and full-body manipulation tasks demonstrate that the six-round CCE setting reduces mean semantic failure counts from 9.26 to 7.83 and 8.85 to 7.65, respectively, while enhancing aggregate visual quality. Furthermore, on the L-IVA benchmark using a shared generation backend, VAEM achieves state-of-the-art perceptible motion amplitude (PAS: 28.0) and motion smoothness (MSS: 87.7) alongside competitive visual quality. Finally, a real-time prototype validates VAEM's compatibility with interactive rendering. These results demonstrate that persistent context evolution offers an effective paradigm for cross-task experience reuse without fine-tuning the underlying video generator. Project page: https://anonymous.4open.science/w/VAEM-04EB.

1 INTRODUCTION

Video avatar generation aims to synthesize high-fidelity human videos driven by external signals, such as speech (Kong et al., 2025; Yang et al., 2025b; Gan et al., 2025), pose sequences (Zhu et al., 2024a; Zhong et al., 2024), text instructions (Tu et al., 2025; Tian et al., 2024; Wan et al., 2025), or combinations of these signals. Such models belong to the *predictor mode* (Figure 1(a)): they make fast, reactive one-step predictions without maintaining a multi-step plan. With the increasing use of video avatars in interactive and open-ended applications, visual fidelity alone is no longer sufficient. A capable video avatar should instead autonomously pursue a high-level goal through multi-step planning and sustain coherent behavior throughout task execution.

Recent work (Jiang et al., 2025; Ding et al., 2025; He et al., 2025b) has begun to develop such video avatars, drawing on the distinction between System 1 and System 2. A video generation model serves as System 1 to render actions, while a multimodal large language model serves as System 2 to plan them (Figure 1(b)). These *simulator mode* methods can produce video sequences guided by multi-step planning. However, their System 2 is episodic and inefficient: it serves only the goal currently being pursued. When a new task starts, the agent does not consolidate past failures into persistent knowledge that improves planning, so similar errors recur across tasks.

A hallmark of genuine System 2 cognition is the ability to learn from past failures and transfer the resulting knowledge across scenarios and tasks. Much as a person does not merely remember a mistake, but extracts a general lesson from it and applies that lesson when facing a new situation, an effective System 2 should abstract generation experience into transferable behavioral principles and reusable skills. We refer to this ability to accumulate, refine, and reuse knowledge across generation episodes as the *evolver mode* (Figure 1(c)). By extending episodic reasoning into persistent self-

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.

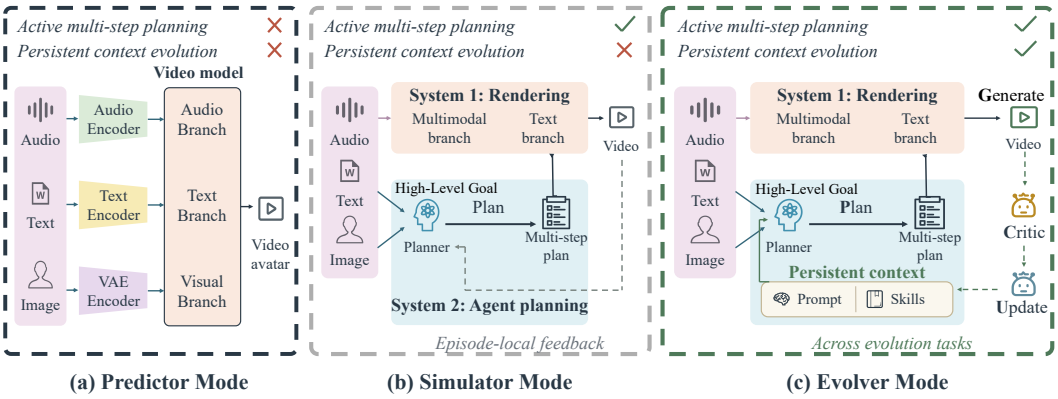


Figure 1: Comparison of predictor, simulator, and evolver modes for video-avatar generation. Only the evolver mode supports both active multi-step planning and persistent self-evolution.

improvement, the evolver mode enables the avatar to plan more effectively for new tasks in different scenes, rather than rediscovering the same solutions for each new task.

To address these challenges, we propose VAEM, a self-evolving framework that equips video avatars with an evolving mind distilled from their own generation histories. Unlike episodic System 2 reasoning that starts largely from scratch for each new goal, VAEM accumulates experience across generation episodes and transforms it into persistent knowledge for future planning. This evolving mind is represented by two complementary components: global behavioral principles encoded in an evolving system prompt, and reusable skills stored in a structured, retrievable skill library. By evolving only the planner's persistent context while keeping the video generation backend frozen, VAEM enables video avatars to continuously improve their planning and prompting across tasks.

To achieve this, VAEM organizes multiple VLM agents into a Plan-Generate-Critique-Update (PGCU) cycle. Given a reference scene and a task goal, a scene-aware planner first reasons about the scene structure and produces executable segment-level action plans and corresponding text prompts. A frozen video avatar generator then renders these plans into video clips. Critique agents evaluate the generated results and provide targeted, fine-grained feedback on planning and execution failures. Based on this feedback, an update agent abstracts recurring failures and successful behavior patterns into global behavioral principles, while distilling task- and scene-relevant experience into reusable skills. For transfer across scenes with different objects, spatial layouts, and interaction constraints, the update agent abstracts instance-specific feedback into general planning rules.

Self-evolution further requires fine-grained feedback that identifies not only whether a failure occurs, but also where and why it occurs. Existing video metrics primarily assess visual quality and motion fidelity, offering limited insight into whether the avatar correctly executes the planned behavior or which segment causes a semantic failure. We therefore introduce the Semantic Behavior Metric (SBM), which evaluates each planned segment along five interpretable dimensions and localizes failures at both the segment and behavior levels. This fine-grained evaluation provides a structured signal for failure attribution, allowing the update agent to distinguish planning-related errors from generation defects and evolve the planner's persistent context accordingly.

We evaluate VAEM on diverse half-body and full-body multi-step tasks to investigate whether context evolution improves semantic goal completion, whether prompt evolution and skill retrieval provide complementary benefits, and how different feedback-distillation strategies affect transfer to held-out tasks. We further evaluate the evolved context on L-IVA (He et al., 2025b) to examine its generalization to an external long-horizon avatar benchmark.

Our main contributions are summarized as follows: (1) We formulate the task of self-evolution for video avatars and present VAEM, a self-evolving framework that equips video avatars with an evolving mind; (2) We rethink traditional approaches to avatar generation and introduce a Plan-Generate-Critique-Update (PGCU) cycle to enable video avatars to operate in *evolver mode*; (3) To provide fine-grained feedback, we introduce the Semantic Behavior Metric (SBM), an evaluation

metric that localizes semantic failures along five interpretable dimensions, providing structured, planning-attributable feedback for context evolution; (4) Extensive experiments on our half-body and full-body manipulation benchmarks and on L-IVA demonstrate our method's superiority in both semantic goal completion and video quality.

2 Related Work

Video Avatar Generation. Avatar video generation spans three modes (Chu et al., 2026) (Figure 1): predictor, simulator, and evolver. In the *predictor mode* (a), motion (Hu, 2024; Zhu et al., 2024b; Yang et al., 2023), audio (Zhang et al., 2023; Xu et al., 2024; Chen et al., 2025b; Kong et al., 2025; Yang et al., 2025b), or text conditions are mapped directly to a clip by a video model (Lin et al., 2025b; Tian et al., 2025; Lin et al., 2025a; Wan et al., 2025; Yang et al., 2025c; Chen et al., 2025a; Contributors, 2025), producing a passive avatar. In the *simulator mode* (b) (Wang et al., 2025a;b; He et al., 2025b), an agent adds explicit planning but re-plans every task until it succeeds (repetitive cost), so it is neither real-time nor self-evolving. In the *evolver mode* (c), which VAEM instantiates, a self-evolving agent learns reusable skills from past trajectories, attaining active intelligence, real-time interaction, and self-evolution at once.

Agent for Video Generation. Modes (b) and (c) make synthesis *agentic*: an agent is placed in front of the generator to add explicit planning through task decomposition and feedback (Yang et al., 2025a; Liu et al., 2024a; Yao et al., 2023; Shinn et al., 2023; Wang et al., 2024a; Park et al., 2023). Multi-agent systems assign specialized roles to story, shot, and synthesis for narrative coherence (Xie et al., 2024; Hu et al., 2024; Yuan et al., 2024), a refinement line runs generate-critique-correct cycles on individual clips (Long et al., 2025; Huang et al., 2024a), and a closed-loop, online-trajectory family studies agentic decision-making inside the generation environment (Wang et al., 2025a;b; He et al., 2025b). These agents, however, perfect a single clip or re-plan each task from scratch, without abstracting failure into transferable, cross-task skills. Closest to us is ORCA (He et al., 2025b), which uses closed-loop planning in the generation environment; VAEM freezes the backend and evolves only the planner's *context* (a system prompt and a retrievable skill library).

Context Optimization, Experience Distillation, and Skill Learning. In language-based agents, persistent *context* provides a lightweight mechanism for encoding reusable planning constraints and experience-derived knowledge without modifying model parameters. Prompt and rule optimizers refine such context from feedback to improve subsequent decisions (Agrawal et al., 2026; Khattab et al., 2024; Yuksekgonul et al., 2025). Learning from experience in agents ranges from storing raw exemplars and reusable workflows (Zheng et al., 2024; Wang et al., 2025c; Xu et al., 2025) to distilling accumulated trajectories into general lessons and structured memory (Zhao et al., 2024; Ouyang et al., 2026; Chhikara et al., 2025; Cai et al., 2025). Recent work further organizes experience into callable and composable *skills* that can be retrieved and reused in new situations (Anthropic, 2026; Ni et al., 2026; Alzubi et al., 2026; Jiang et al., 2026). VAEM combines these directions in goal-oriented avatar video generation. It distills generation histories into two complementary forms of persistent knowledge: general behavioral principles encoded in the system prompt and reusable procedural skills stored in a structured skill library. Unlike prior methods developed primarily for text, web, or code domains, VAEM learns from semantic failures involving physical actions, object-state transitions, and cross-clip continuity under a frozen video generation backend.

Automatic Evaluation and VLM-as-Critic. Reliable automatic evaluation is essential for enabling self-evolving systems. Distribution-based metrics such as FID and FVD (Heusel et al., 2017; Unterthiner et al., 2018) primarily assess distribution-level visual quality, but cannot identify or localize action failures in individual videos. Recent multi-dimensional benchmarks provide more interpretable assessments of generated-video quality (Huang et al., 2024b; Liu et al., 2024b; Huang et al., 2026; Han et al., 2025), with several studies further extending evaluation beyond appearance to physical plausibility, commonsense consistency, and semantic correctness (Zheng et al., 2025; Wang et al., 2024b; Bansal et al., 2025; Motamed et al., 2026). Meanwhile, advances in VLM-based judging (Zheng et al., 2023) have enabled video critics and reward models to provide fine-grained feedback for both evaluation and generator optimization (He et al., 2025a; Xu et al., 2026; Zhang et al., 2025b; Liu et al., 2025; Zhang et al., 2025a). Closest to our setting is SPOTLIGHT (Chinchure et al., 2025), which localizes generation errors rather than assigning only holistic scores, and reports strong zero-shot performance with Gemini 2.5 Pro as well as further improvements through multi-

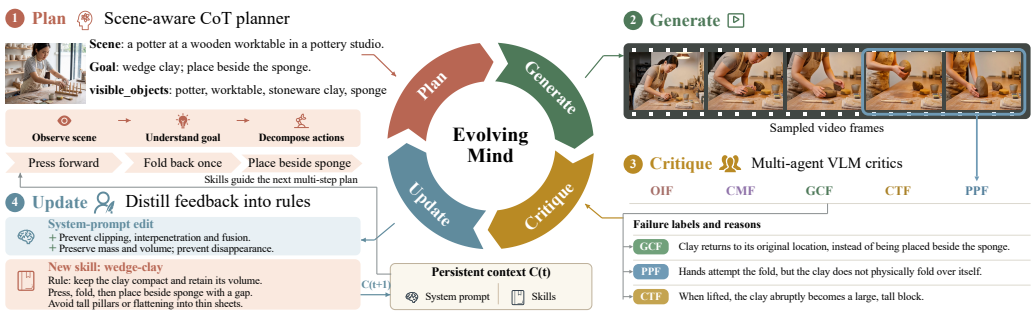


Figure 2: Overview of VAEM. Skill retrieval, scene-aware planning, and video generation form the generation path. During context evolution, VLM critics localize semantic failures and the update agent uses plans and critique feedback to infer behavioral principles and reusable skills. The clay-manipulation example illustrates this PGCU cycle; the displayed frames are illustrative, with the complete-video critique protocol detailed in Appendix A.4. At inference, the final context is fixed and reused for skill retrieval, a single planning pass, and segmented generation.

agent evaluation. Building on these advances, we introduce the Semantic Behavior Metric (SBM), which measures five categories of segment-level semantic failures—physical plausibility, commonsense consistency, goal completion, cross-clip continuity, and object integrity—denoted as PPF, CMF, GCF, CTF, and OIF, respectively. Their sum is reported as TPF@SBM. The resulting localized critiques provide structured supervision for prompt and skill updates, while VMBench (Ling et al., 2025) serves as a complementary measure of visual quality.

3 Method

Given a reference image I_{ref} and a high-level goal g, goal-oriented avatar generation aims to synthesize a video sequence $V = (v_1, \ldots, v_K)$ that fulfills g while preserving identity, scene layout, and object-state continuity across K clips. VAEM uses a planner π and a frozen video generator G. To retain execution experience beyond individual tasks, it optimizes persistent planner context $\mathcal{C} = (P_{\text{sys}}, \mathcal{S})$: a system prompt P_{sys} encoding global behavioral principles and a skill library $\mathcal{S}$ storing reusable procedural knowledge. This makes $\mathcal{C}$ the optimization variable, with all model parameters fixed.

As illustrated in Figure 2, the Plan-Generate-Critique-Update (PGCU) cycle performs gradient-free optimization in textual context space. During offline evolution, semantic critiques guide revisions to rules and skills. At inference, the selected context $\mathcal{C}^\star$ is frozen and reused for single-pass planning without online context updates or replanning.

3.1 Context-Conditioned Avatar Generation

For a target task (I_{ref}, g), generation proceeds through skill retrieval, scene-aware planning, and anchored segmented execution.

The system prompt supplies general constraints, while skills provide reusable task procedures. Each skill contains scene tags, a task pattern, planning rules, and anti-patterns. VAEM loads global skills and retrieves scene-specific skills matched to the scene and goal; $\mathcal{S}_{\text{rel}}$ denotes their combined active set:

$$\mathcal{S}_{\text{rel}} = \text{Retrieve}(\mathcal{S}, I_{\text{ref}}, g), \qquad p = \pi(I_{\text{ref}}, g; P_{\text{sys}}, \mathcal{S}_{\text{rel}}). \tag{1}$$

The retrieved rules enter a dedicated system-prompt slot alongside the planner's role, planning constraints, and JSON output specification. This conditions new plans on transferable experience, guiding action descriptions toward the backend's execution capabilities. Appendices A.3 and A.6 detail the prompt and skill structures.

The planner produces $p = (p_1, \ldots, p_K)$ through three stages of chain-of-thought (CoT) reasoning, progressively grounding the task and controlling the complexity assigned to each generated clip. **Observe the scene:** identify visible objects, human pose, hand positions, gaze, and spatial layout to

ground possible actions in I_{ref}. **Understand the goal:** identify target state transitions, action order, preconditions, and completion conditions, distinguishing required changes from states that should persist. **Decompose actions:** assign one main physical transition to each segment, with explicit camera constraints and continuity declarations, to avoid overloading the generator with simultaneous state changes.

Each p_j specifies the segment index, main action, participating and static objects, camera constraint, action description, and continuity declaration. In the clay example of Figure 2, explicitly separating pressing, folding, and final placement makes the required transitions clearer than a generic instruction to *work the clay*.

The frozen image-to-video (I2V) backend G renders each segment from an anchor frame a_j:

$$\begin{aligned} &a_1 = I_{\text{ref}}, \qquad v_j = G(a_j, p_j), \\ &a_j = \text{SelectAnchor}(I_{\text{ref}}, \text{LastFrame}(v_{j-1}), p_j), \quad j > 1. \end{aligned} \tag{2}$$

The continuity declaration selects either the preceding clip's last frame, carrying forward its object states, or I_{ref}, re-anchoring reference appearance when the plan does not require the preceding state. This choice makes the conditioning dependency explicit. Camera and pose constraints, together with tracking each object's original and new locations, aim to limit viewpoint jumps and accumulated state drift. We denote the composed sequence-level procedure by $V = \text{Render}(G, I_{\text{ref}}, p)$. A coherent plan can still fail under stochastic rendering, motivating feedback about which planning constraints should change.

3.2 Critique-Guided Context Evolution

Context-space optimization requires more than a scalar quality score: the updater needs localized evidence linking execution failures to rules it can revise. VAEM combines semantic failure counting with natural-language credit assignment over the planner's persistent context.

Five specialized VLM critics, collectively denoted by E, assess physical plausibility (PPF), commonsense consistency (CMF), goal completion (GCF), cross-clip continuity (CTF), and object integrity (OIF) (Zheng et al., 2023; He et al., 2024). Given the reference image, goal, plan, and generated sequence, they return records containing an error description and localization through segment indices and timestamps. GCF also checks omitted goal steps; CTF focuses on adjacent clips.

For an evaluated trajectory (x, p, V), where $x = (I_{\text{ref}}, g)$, let $\mathcal{F}_d(x, p, V)$ contain the failure records for dimension $d \in \mathcal{D}$, where $\mathcal{D}$ is the set of five dimensions above. Records are deduplicated by normalized description within each dimension and video. The Semantic Behavior Metric (SBM) is

$$\text{SBM}(x, p, V) = \sum_{d \in \mathcal{D}} |\mathcal{F}_d(x, p, V)|. \tag{3}$$

Multiple failures may occur within one segment and dimension, and omitted steps count even when no corresponding segment was generated. The scalar count ranks executions; the underlying records retain the diagnostic evidence needed for revision. We report mean per-dimension counts and their mean total, TPF@SBM, alongside VMBench visual-quality scores (Ling et al., 2025).

The update agent R reads plans and critique records to infer actionable context edits. Localized descriptions provide a qualitative analogue of an optimization direction in text space: they identify what failed and suggest which rules could address it. This credit assignment is performed through natural-language revision, without gradients, separate attribution labels, or a programmatic failure filter. All reported failures contribute to SBM. The updater is instructed to focus on semantic planning corrections; for example, ambiguous placement can motivate an explicit destination rule, whereas texture noise alone does not establish such a correction.

Global rule distillation abstracts recurring failures into system-prompt principles while retaining useful constraints (Zhang et al., 2026), such as requiring a moved object to leave its original location empty. Procedural skill synthesis stores task-specific repairs and successful patterns as reusable planning rules and anti-patterns (Zhao et al., 2024), such as separating grasping, carrying, and placing clay beside a sponge. These structured records are retrieved for future tasks.

At evolution round t, let $\mathcal{C}^{(t)} = (P_{\text{sys}}^{(t)}, \mathcal{S}^{(t)})$. Absorbing skill retrieval into π, the PGCU cycle is

$$\begin{aligned} p^{(t)} &= \pi(I_{\text{ref}}, g; \mathcal{C}^{(t)}), & V^{(t)} &= \text{Render}(G, I_{\text{ref}}, p^{(t)}), \\ f^{(t)} &= E(I_{\text{ref}}, g, p^{(t)}, V^{(t)}), & \mathcal{C}^{(t+1)} &= R(\mathcal{C}^{(t)}, p^{(t)}, f^{(t)}). \end{aligned} \tag{4}$$

Here $f^{(t)}$ contains the localized critique records. The last expression summarizes the revision interface; the following strategies determine how plans and feedback are aggregated across tasks.

Single-plan Holistic Context Evolution (**HCE**) samples one plan per evolution task. After all tasks are rendered and critiqued, R_{hce} jointly revises the current context from the collected plans and feedback, preserving useful prior constraints.

Multi-plan Contrastive Context Evolution (**CCE**) samples $L = 4$ candidate plans $\{p_{i,\ell}\}_{\ell=1}^{L}$ per task $x_i = (I_i, g_i)$ under the same context $\mathcal{C}^{(t)}$. Rendering and critique yield $V_{i,\ell}$, feedback $f_{i,\ell}$, and $s_{i,\ell} = \text{SBM}(x_i, p_{i,\ell}, V_{i,\ell})$ for each candidate. CCE selects the best and worst candidates by their total failure counts:

$$\ell_i^+ = \arg\min_{\ell} s_{i,\ell}, \qquad \ell_i^- = \arg\max_{\ell} s_{i,\ell}. \tag{5}$$

The corresponding plans p_i^+, p_i^- and feedback f_i^+, f_i^- guide a task-specific edit $\Delta_i = R_{\text{cce}}(\mathcal{C}^{(t)}, p_i^+, p_i^-, f_i^+, f_i^-)$. All task edits are then merged into $\mathcal{C}^{(t+1)}$. Comparing alternatives for the same goal gives the updater a task-matched basis for retaining effective procedures and identifying actionable prompt ambiguities. This contrastive credit assignment aims to distinguish planning choices that warrant revision from incidental rendering defects; attribution remains a prompt-guided judgment. Appendix A.4 details the critique and revision procedures.

Evolution and inference. Offline evolution repeats trajectory collection and context revision. HCE and component-ablation runs use 12 and 6 rounds, respectively; the three- and six-round CCE evaluation settings are distinguished in Appendix A.2. VAEM freezes a selected checkpoint $\mathcal{C}^\star$ for testing: a new task uses skill retrieval when enabled, one planner call, and segmented generation. Test-time scoring does not update the context. In deployment, inference trajectories—including task inputs, plans, generated videos, and any subsequently collected critiques—can be archived for later offline PGCU rounds to refine the system prompt and skill library. Such reuse is separate from the held-out evaluation reported here. Algorithms 1, 2, and 3 provide the complete procedures.

4 Experiments

We first investigate whether context evolution improves semantic goal completion under an I2V backend. We then perform ablation studies to disentangle the individual impacts of system prompt evolution (general principles) and dynamic skill retrieval.

4.1 Setup

Datasets. We build a multi-scene manipulation benchmark over two families of avatar reference images. The half-body tabletop family covers ten desk-scale scenes (e.g., craft studio, office desk, ceramic cafe); the full-body family covers ten scenes with a larger motion space (e.g., fitness studio, kitchen, warehouse floor). For each scene, a single reference image is paired with multiple natural-language interaction goals. The goals are generated by VLMs and screened by humans to ensure that each can be decomposed into 3–6 semantically explicit action steps. Each (reference image, goal) pair is a task instance. Each scene contains 10 evolution tasks and 20 held-out test tasks, partitioned by task semantics so that highly similar descriptions do not straddle the split, yielding 200 test task instances per family.

Settings. We compare four settings: Control (base system prompt), Full, w/o skill, and w/o prompt. Unless otherwise stated, the main ablation uses multi-plan Contrastive Context Evolution (CCE), with plan samples size of 4 for 6 rounds. Table 1 retains both CCE settings: the exact third-round prompt (v3), with skill retrieval disabled, and the earlier result selected from a six-round run. Single-plan Holistic Context Evolution (HCE) uses plan samples size of 1 for 12 rounds. All test tasks use one plan; four-candidate sampling is confined to CCE evolution. The third-round CCE rerun uses the 20 selected held-out tasks per scene and supplies each complete generated video to all five critics, without sampled-frame fallback. GPT-5.4 serves as the planner except on L-IVA, where

Table 1: Semantic-completion and visual-quality results on the held-out half-body and full-body test sets. SBM is reported as the mean semantic-failure count (↓), with TPF denoting the sum of its five dimensions. VMBench scores are multiplied by 100 (↑), and "total" is the mean of its five dimensions. Here N denotes the evolution-round setting: HCE uses one plan per goal, while CCE uses four candidates with results retained for N=3 and N=6. Every method uses one plan at test time. Totals are computed before rounding; bold denotes the best displayed values within each framing.

		SBM (failure count, ↓)						**VMBench** (↑)					
Avatar framing	Method	PPF	CMF	GCF	CTF	OIF	TPF	PAS	OIS	TCS	MSS	CAS	total
Half-body	Control	2.25	1.63	1.60	1.81	1.97	9.26	**22.3**	84.8	96.6	85.8	55.7	69.0
	HCE (N=12)	2.13	1.42	1.46	1.62	1.82	8.45	21.6	84.1	**97.8**	86.7	55.7	69.2
	CCE (N=3)	2.12	1.36	1.47	1.54	1.86	8.34	22.2	**86.1**	96.8	**87.3**	55.9	69.6
	CCE (N=6)	**2.04**	**1.26**	**1.30**	**1.50**	**1.73**	**7.83**	22.0	86.0	97.4	87.2	**56.5**	**69.8**
Full-body	Control	2.10	1.69	1.64	1.73	1.69	8.85	**24.4**	66.4	96.3	82.1	54.6	64.8
	HCE (N=12)	2.12	1.37	**1.37**	1.74	1.65	8.24	22.4	67.6	97.3	82.1	55.2	64.9
	CCE (N=3)	2.03	**1.28**	1.41	1.56	1.65	7.93	22.5	**70.6**	97.6	81.2	55.2	65.4
	CCE (N=6)	**1.88**	1.38	1.38	**1.51**	**1.50**	**7.65**	23.8	70.5	**98.1**	**82.8**	**56.0**	**66.2**

Table 2: Comparison of planning methods on L-IVA with a shared frozen video-generation backend.

	SBM (failure count, ↓)						**VMBench** (↑)					
Method	PPF	CMF	GCF	CTF	OIF	TPF	PAS	OIS	TCS	MSS	CAS	total
Open-Loop	2.43	1.76	1.80	1.98	2.09	10.06	26.2	60.6	98.3	85.9	50.5	64.31
ORCA (He et al., 2025b)	2.30	1.69	1.74	1.73	1.98	9.44	20.3	68.7	96.4	85.6	51.6	64.48
Reactive (Yao et al., 2023)	**2.08**	1.71	1.65	**1.59**	**1.70**	**8.73**	22.2	**69.8**	97.9	86.2	**52.8**	**65.78**
VAGEN (Wang et al., 2025a)	2.25	1.70	1.68	1.93	2.09	9.65	26.3	62.6	**98.4**	86.0	50.8	64.82
VAEM	2.15	**1.61**	**1.51**	1.61	1.99	8.87	**28.0**	61.2	97.0	**87.7**	52.0	65.18

we use Gemini-2.5-Flash. Gemini-3.1-Pro-Preview serves as the backbone for the five specialized critique agents and the update agent. The video backend is fixed across all settings: Wan2.1-I2V-14B-480P with LightX2V acceleration and a FusionX LoRA (Wan et al., 2025; Contributors, 2025), using 8 sampling steps, guidance 1.0, and 81 frames per segment. The third-round CCE rerun uses random generation seeds rather than seeds matched to the earlier baselines; the comparison therefore includes generation variability.

Metrics. We report semantic failures with SBM (↓): physical plausibility (PPF), commonsense consistency (CMF), goal completion (GCF), cross-clip continuity (CTF), object integrity (OIF), and their sum (TPF). VMBench (Ling et al., 2025) measures visual quality (↑); total@VM averages PAS/OIS/TCS/MSS/CAS. Although SPOTLIGHT (Chinchure et al., 2025) shows that multi-agent VLM evaluation can agree with human experts, we further validate our task-specific metric by comparing automated SBM judgments with human annotations; the results in Appendix A.5 show close alignment with human judgments.

4.2 Quantitative Analysis

Main comparison. As shown in Table 1, Contrastive Context Evolution (CCE) consistently outperforms both Control and Holistic Context Evolution (HCE) on aggregate metrics across half-body and full-body benchmarks. Crucially, under an **equal trajectory-evaluation budget**—three-round CCE and twelve-round HCE both evaluate 12 trajectories per evolution task (4×3 versus 1×12)—CCE (N=3) already surpasses HCE across all aggregate metrics (e.g., TPF of 8.34 vs. 8.45 on half-body tasks). The six-round CCE setting achieves substantial reductions in aggregate semantic failures (TPF), lowering mean error counts relative to Control by **15.4%** (from 9.26 to 7.83) on half-body tasks and **13.6%** (from 8.85 to 7.65) on full-body tasks, while concurrently boosting overall visual quality (total@VM up to 69.8 and 66.2). These gains are particularly striking in reasoning-heavy dimensions—for instance, CMF drops by up to **22.7%** and GCF by **18.8%** on half-body tasks.

These results support the effectiveness of contrasting execution trajectories for rule distillation, with gains over holistic feedback already evident at the same trajectory-evaluation budget.

External transfer (L-IVA). Table 2 evaluates zero-shot transfer performance on L-IVA under a shared frozen backend. VAEM achieves state-of-the-art perceptible motion amplitude (PAS: 28.0) and smoothness (MSS: 87.7), alongside the lowest failure counts in commonsense consistency (CMF: 1.61) and goal completion (GCF: 1.51). These gains confirm that the evolved context generalizes robustly to disjoint, long-horizon task distributions, effectively guiding both semantic execution and visual motion fidelity without requiring backend fine-tuning.

Test-time planning efficiency. Table 3 highlights the distinct operational efficiency of our approach. While online replanning baselines like ORCA incur high computational overhead via iterative trial-and-error (averaging 5.59 calls and 2.02 retries per task), VAEM produces a coherent plan averaging 4.66 steps in a **single planning pass (1.00 call, 0 retries)**. This demonstrates that offline context evolution successfully internalizes execution constraints, eliminating the need for costly test-time correction while preserving execution rigor.

Table 3: **Test-time planning overhead on L-IVA.** Mean planner calls, accepted plan steps, and retries per case. VAEM produces the full plan in one call without retries.

Method	Calls ↓	Steps	Retries ↓
ORCA (He et al., 2025b)	5.59	3.57	2.02
Open-Loop	6.15	6.15	0.00
Reactive (Yao et al., 2023)	4.12	4.12	0.00
VAGEN (Wang et al., 2025a)	5.44	5.44	0.00
VAEM	**1.00**	4.66	0.00

4.3 Ablation Analysis

Table 4 validates the complementary roles of prompt evolution and skill retrieval. While prompt evolution primarily governs global physical and logical consistency (PPF, CMF, GCF), skill retrieval provides targeted procedural guidance that noticeably improves cross-clip continuity (CTF). Their combination proves especially effective for object permanence: the full framework achieves an OIF of 1.50—representing reductions of **9.1%** and **6.3%** in object-integrity failures compared to *w/o skill* and *w/o prompt*, respectively. This demonstrates that global principles and retrieved procedural skills synergize to prevent accumulated state degradation in multi-step execution.

4.4 Qualitative Analysis

Figure 3 provides qualitative comparisons on half-body and full-body tasks using the identical frozen backend. In the tabletop task (a), Control suffers from camera handling errors and visual distortion, whereas VAEM faithfully executes the multi-step sequence—successfully inserting intermediate ring-light adjustments without disrupting the surrounding setup. Similarly, in the full-body placement task (b), Control exhibits object deformation and inaccurate placement, while VAEM maintains rigid object shape and achieves precise spatial alignment. These visual results illustrate how explicit action decomposition and state-tracking constraints mitigate state drift across multi-clip generation.

Table 4: Component ablation using the six-round CCE setting on full-body tasks, averaged over ten scenes. *w/o skill* disables skill retrieval; *w/o prompt* disables prompt evolution. VMBench scores are multiplied by 100, and "total" is the mean of PAS, OIS, TCS, MSS, and CAS. Bold and underlined values indicate the best and second-best displayed scores, respectively.

	SBM (failure count, ↓)						**VMBench** (↑)					
Setting	PPF	CMF	GCF	CTF	OIF	TPF	PAS	OIS	TCS	MSS	CAS	total
Control	2.10	1.69	1.64	1.73	1.69	8.85	**24.4**	66.4	96.3	82.1	54.6	64.8
w/o skill	**1.81**	**1.31**	**1.35**	1.60	1.65	<u>7.72</u>	23.3	68.4	**98.1**	<u>82.9</u>	55.5	65.7
w/o prompt	<u>1.87</u>	1.42	1.42	**1.48**	<u>1.60</u>	7.78	23.4	<u>69.3</u>	<u>97.3</u>	**83.8**	<u>55.9</u>	<u>65.9</u>
Full	1.88	<u>1.38</u>	<u>1.38</u>	<u>1.51</u>	**1.50**	**7.65**	<u>23.8</u>	**70.5**	**98.1**	82.8	**56.0**	**66.2**

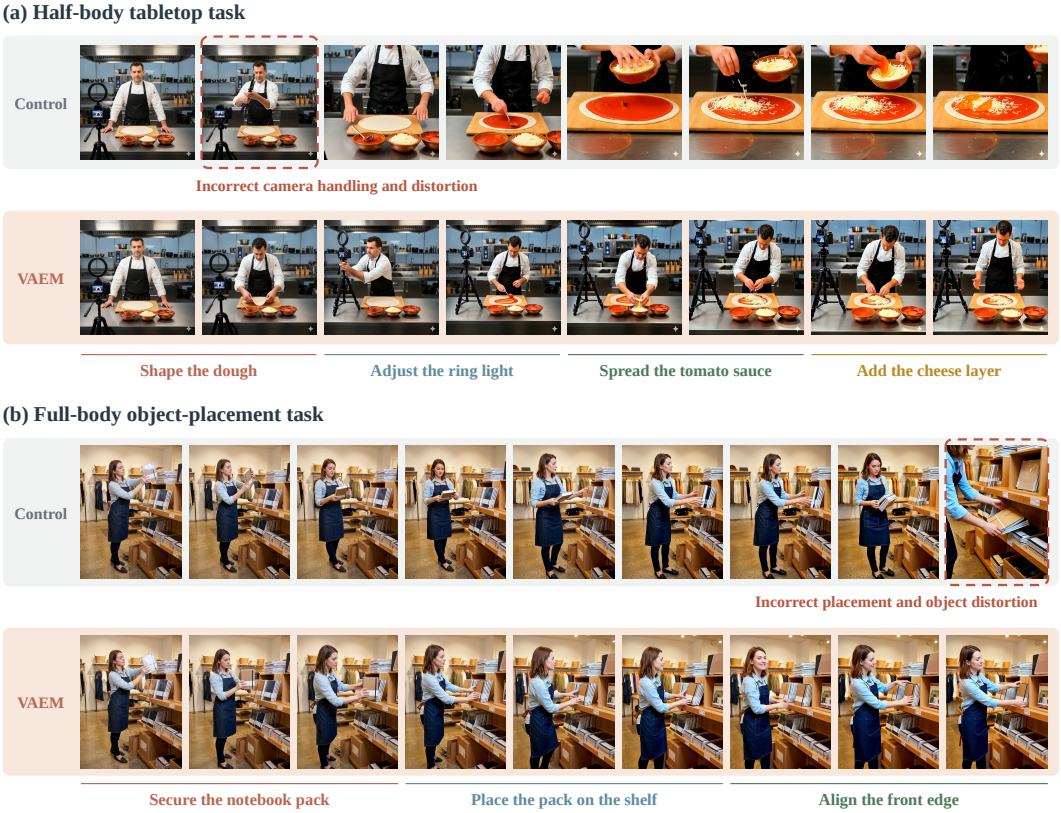


Figure 3: Qualitative comparison on two representative half/full-body tasks using the same frozen video-generation backend. Control rows (gray) exhibit camera-handling errors, object deformation, and incorrect placement, highlighted by red boxes. VAEM rows (orange) follow the labeled atomic actions more consistently and better preserve intermediate object states across segments.

5 Conclusion

We presented VAEM, a framework equipping video avatars with persistent planning through a closed-loop PGCU cycle. Guided by the fine-grained Semantic Behavior Metric (SBM), we explore the self-evolution efficiency of Holistic (HCE) and Contrastive Context Evolution (CCE) over same renderer. Experimental results show that CCE yields significantly higher evolution efficiency and semantic quality gains. Component ablations confirm the complementary contributions of prompt evolution and skill retrieval, establishing persistent context evolution as a lightweight route from episodic planning to reusable avatar behavior.

AI USE STATEMENT

The experimental system uses generative AI for planning, video generation, semantic critique, and context updates. Benchmark goals are generated by VLMs and filtered by humans, as described in Section 4.1. Generative AI was also used to assist with language editing and manuscript preparation. The authors are responsible for verifying AI-assisted content and the accuracy of the final manuscript.

ETHICS STATEMENT

VAEM enables goal-oriented video avatar generation and shares the potential risks of video synthesis, including deceptive media, impersonation, and biases inherited from pretrained models and their training data. Generated videos should be clearly identified as synthetic. The collection and use of reference images and videos, as well as system deployment, should respect privacy, consent where applicable, and the licenses of source materials and pretrained models. Our human study evaluates the physical plausibility and task completion of generated videos; its protocol and aggregate results are described in Appendix A.5.

REPRODUCIBILITY STATEMENT

Section 4.1 specifies benchmark splits, model backbones, evolution settings, and generation parameters. Appendix A.2 provides pseudocode for PGCU, HCE, and CCE. Appendices A.3 and A.6 describe the planner prompt structure and skill representation, while Appendices A.4 and A.5 detail the critique procedure, metric aggregation, and human-evaluation protocol. We will release the code as open source to support reproducibility and further research.

REFERENCES

Lakshya A. Agrawal, Shangyin Tan, Dilara Soylu, Noah Ziems, Rishi Khare, Krista Opsahl-Ong, Arnav Singhvi, Herumb Shandilya, Michael J. Ryan, Meng Jiang, Christopher Potts, Koushik Sen, Alexandros G. Dimakis, Ion Stoica, Dan Klein, Matei Zaharia, and Omar Khattab. Gepa: Reflective prompt evolution can outperform reinforcement learning. In *ICLR*, 2026.

Salaheddin Alzubi, Noah Provenzano, Jaydon Bingham, Weiyuan Chen, and Tu Vu. Evoskill: Automated skill discovery for multi-agent systems, 2026. URL https://arxiv.org/abs/2603.02766.

Anthropic. What are skills?, 2026. URL https://support.claude.com/en/articles/12512176-what-are-skills.

Hritik Bansal, Zongyu Lin, Tianyi Xie, Zeshun Zong, Michal Yarom, Yonatan Bitton, Chenfanfu Jiang, Yizhou Sun, Kai-Wei Chang, and Aditya Grover. Videophy: Evaluating physical commonsense for video generation. In *ICLR*, 2025.

Yuzheng Cai, Siqi Cai, Yuchen Shi, Zihan Xu, Lichao Chen, Yulei Qin, Xiaoyu Tan, Gang Li, Zongyi Li, Haojia Lin, Yong Mao, Ke Li, and Xing Sun. Training-free group relative policy optimization, 2025. URL https://arxiv.org/abs/2510.08191.

Yi Chen, Sen Liang, Zixiang Zhou, Ziyao Huang, Yifeng Ma, Junshu Tang, Qin Lin, Yuan Zhou, and Qinglin Lu. Hunyuanvideo-avatar: High-fidelity audio-driven human animation for multiple characters. *arXiv preprint arXiv:2505.20156*, 2025a.

Zhiyuan Chen, Jiajiong Cao, Zhiquan Chen, Yuming Li, and Chenguang Ma. Echomimic: Lifelike audio-driven portrait animations through editable landmark conditions. In *AAAI*, volume 39, pp. 2403–2410, 2025b.

Prateek Chhikara, Dev Khant, Saket Aryan, Taranjeet Singh, and Deshraj Yadav. Mem0: Building production-ready ai agents with scalable long-term memory. In *ECAI*, 2025.

Aditya Chinchure, Sahithya Ravi, Pushkar Shukla, Vered Shwartz, and Leonid Sigal. Spotlight: Identifying and localizing video generation errors using vlms, 2025. URL https://arxiv.org/abs/2511.18102.

Meng Chu, Xuan Billy Zhang, Kevin Qinghong Lin, Lingdong Kong, Jize Zhang, Teng Tu, Weijian Ma, Ziqi Huang, Senqiao Yang, Wei Huang, Yeying Jin, Zhefan Rao, Jinhui Ye, Xinyu Lin, Xichen Zhang, Qisheng Hu, Shuai Yang, Leyang Shen, Wei Chow, Yifei Dong, Fengyi Wu, Quanyu Long, Bin Xia, Shaozuo Yu, Mingkang Zhu, Wenhu Zhang, Jiehui Huang, Haokun Gui, Runyi Li, Chenyu Tang, Dong Huang, Xuhang Chen, Rui Liu, Chengzu Li, Shiyi Du, Xu Huang, Haoxuan Che, Long Chen, Qifeng Chen, Wenya Wang, Wenxuan Zhang, Xiaojuan Qi, Yang Deng, Yanwei Li, Mike Zheng Shou, Zhi-Qi Cheng, See-Kiong Ng, Ziwei Liu, Philip Torr, and Jiaya Jia. Agentic world modeling: Foundations, capabilities, laws, and beyond. 2026. URL https://arxiv.org/abs/2604.22748.

LightX2V Contributors. Lightx2v: Light video generation inference framework. https://github.com/ModelTC/lightx2v, 2025.

Yikang Ding, Jiwen Liu, Wenyuan Zhang, Zekun Wang, Wentao Hu, Liyuan Cui, Mingming Lao, Yingchao Shao, Hui Liu, Xiaohan Li, Ming Chen, Xiaoqiang Liu, Yu-Shen Liu, and Pengfei Wan. Kling-avatar: Grounding multimodal instructions for cascaded long-duration avatar animation synthesis. *ArXiv*, abs/2509.09595, 2025. URL https://api.semanticscholar.org/CorpusID:281252174.

Qijun Gan, Ruizi Yang, Jianke Zhu, Shaofei Xue, and Steven Hoi. Omniavatar: Efficient audio-driven avatar video generation with adaptive body animation. *arXiv preprint arXiv:2506.18866*, 2025.

Hui Han, Siyuan Li, Jiaqi Chen, Yiwen Yuan, Yuling Wu, Chak Tou Leong, Hanwen Du, Junchen Fu, Youhua Li, Jie Zhang, Chi Zhang, Li jia Li, and Yongxin Ni. Video-bench: Human-aligned video generation benchmark. In *CVPR*, 2025.

Xuan He, Dongfu Jiang, Ge Zhang, Max Ku, Achint Soni, Sherman Siu, Haonan Chen, Abhranil Chandra, Ziyan Jiang, Aaran Arulraj, Kai Wang, Quy Duc Do, Yuansheng Ni, Bohan Lyu, Yaswanth Narsupalli, Rongqi Fan, Zhiheng Lyu, Yuchen Lin, and Wenhu Chen. Videoscore: Building automatic metrics to simulate fine-grained human feedback for video generation. In *EMNLP*, 2024.

Xuan He, Dongfu Jiang, Ping Nie, Minghao Liu, Zhengxuan Jiang, Mingyi Su, Wentao Ma, Junru Lin, Chun Ye, Yi Lu, Keming Wu, Benjamin Schneider, Quy Duc Do, Zhuofeng Li, Yiming Jia, Yuxuan Zhang, Guo Cheng, Haozhe Wang, Wangchunshu Zhou, Qunshu Lin, Yuanxing Zhang, Ge Zhang, Wenhao Huang, and Wenhu Chen. Videoscore2: Think before you score in generative video evaluation, 2025a. URL https://arxiv.org/abs/2509.22799.

Xuanhua He, Tianyu Yang, Ke Cao, Ruiqi Wu, Cheng Meng, Yong Zhang, Zhuoliang Kang, Xiaoming Wei, and Qifeng Chen. Active intelligence in video avatars via closed-loop world modeling, 2025b. URL https://arxiv.org/abs/2512.20615.

Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *Advances in neural information processing systems*, 30, 2017.

Li Hu. Animate anyone: Consistent and controllable image-to-video synthesis for character animation. In *CVPR*, 2024.

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.

Kaiyi Huang, Yukun Huang, Xuefei Ning, Zinan Lin, Yu Wang, and Xihui Liu. Genmac: compositional text-to-video generation with multi-agent collaboration. *arXiv preprint arXiv:2412.04440*, 2024a.

Yubo Huang, Hailong Guo, Fangtai Wu, Shifeng Zhang, Shijie Huang, Qijun Gan, Lin Liu, Sirui Zhao, Enhong Chen, Jiaming Liu, and Steven Hoi. Live avatar: Streaming real-time audio-driven avatar generation with infinite length, 2025. URL https://arxiv.org/abs/2512.04677.

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.

Ziqi Huang, Fan Zhang, Xiaojie Xu, Yinan He, Jiashuo Yu, Ziyue Dong, Qianli Ma, Nattapol Chanpaisit, Chenyang Si, Yuming Jiang, Yaohui Wang, Xinyuan Chen, Ying-Cong Chen, Limin Wang, Dahua Lin, Yu Qiao, and Ziwei Liu. VBench++: Comprehensive and versatile benchmark suite for video generative models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2026.

Guanyu Jiang, Zhaochen Su, Xiaoye Qu, and Yi R. Fung. Xskill: Continual learning from experience and skills in multimodal agents. In *ICML*, 2026.

Jianwen Jiang, Weihong Zeng, Zerong Zheng, Jiaqi Yang, Chao Liang, Wang Liao, Han Liang, Yuan Zhang, and Mingyuan Gao. Omnihuman-1.5: Instilling an active mind in avatars via cognitive simulation. *arXiv preprint arXiv:2508.19209*, 2025.

Omar Khattab, Arnav Singhvi, Paridhi Maheshwari, Zhiyuan Zhang, Keshav Santhanam, Sri Vardhamanan A, Saiful Haq, Ashutosh Sharma, Thomas T. Joshi, Hanna Moazam, Heather Miller, Matei Zaharia, and Christopher Potts. DSPy: Compiling declarative language model calls into state-of-the-art pipelines. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=sY5N0zY5Od.

Zhe Kong, Feng Gao, Yong Zhang, Zhuoliang Kang, Xiaoming Wei, Xunliang Cai, Guanying Chen, and Wenhan Luo. Let them talk: Audio-driven multi-person conversational video generation. *arXiv preprint arXiv:2505.22647*, 2025.

Gaojie Lin, Jianwen Jiang, Chao Liang, Tianyun Zhong, Jiaqi Yang, Zerong Zheng, and Yanbo Zheng. Cyberhost: A one-stage diffusion framework for audio-driven talking body generation. In *ICLR*, 2025a.

Gaojie Lin, Jianwen Jiang, Jiaqi Yang, Zerong Zheng, and Chao Liang. Omnihuman-1: Rethinking the scaling-up of one-stage conditioned human animation models. In *ICCV*, 2025b.

Xinran Ling, Chen Zhu, Meiqi Wu, Hangyu Li, Xiaokun Feng, Cundian Yang, Aiming Hao, Jiashu Zhu, Jiahong Wu, and Xiangxiang Chu. VMBench: A benchmark for perception-aligned video motion generation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 13087–13098, 2025.

Haotian Liu, Chunyuan Li, Yuheng Li, Bo Li, Yuanhan Zhang, Sheng Shen, and Yong Jae Lee. Llava-next: Improved reasoning, ocr, and world knowledge, 1 2024a. URL https://llava-vl.github.io/blog/2024-01-30-llava-next/.

Jie Liu, Gongye Liu, Jiajun Liang, Ziyang Yuan, Xiaokun Liu, Mingwu Zheng, Xiele Wu, Qiulin Wang, Menghan Xia, Xintao Wang, Xiaohong Liu, Fei Yang, Pengfei Wan, Di Zhang, Kun Gai, Yujiu Yang, and Wanli Ouyang. Improving video generation with human feedback. In *NeurIPS*, 2025.

Yaofang Liu, Xiaodong Cun, Xuebo Liu, Xintao Wang, Yong Zhang, Haoxin Chen, Yang Liu, Tieyong Zeng, Raymond Chan, and Ying Shan. Evalcrafter: Benchmarking and evaluating large video generation models. In *CVPR*, 2024b.

Do Xuan Long, Xingchen Wan, Hootan Nakhost, Chen-Yu Lee, Tomas Pfister, and Sercan Ö Arık. Vista: A test-time self-improving video generation agent. *arXiv preprint arXiv:2510.15831*, 2025.

Saman Motamed, Laura Culp, Kevin Swersky, Priyank Jaini, and Robert Geirhos. Do generative video models understand physical principles? In *WACV*, 2026.

Jingwei Ni, Yihao Liu, Xinpeng Liu, Yutao Sun, Mengyu Zhou, Pengyu Cheng, Dexin Wang, Erchao Zhao, Xiaoxi Jiang, and Guanjun Jiang. Trace2skill: Distill trajectory-local lessons into transferable agent skills, 2026. URL https://arxiv.org/abs/2603.25158.

Siru Ouyang, Jun Yan, I-Hung Hsu, Yanfei Chen, Ke Jiang, Zifeng Wang, Rujun Han, Long Le, Samira Daruki, Xiangru Tang, Vishy Tirumalashetty, George Lee, Mahsan Rofouei, Hangfei Lin, Jiawei Han, Chen-Yu Lee, and Tomas Pfister. Reasoningbank: Scaling agent self-evolving with reasoning memory. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=jL7fwchScm.

Joon Sung Park, Joseph C. O'Brien, Carrie J. Cai, Meredith Ringel Morris, Percy Liang, and Michael S. Bernstein. Generative agents: Interactive simulacra of human behavior. In *CHI*, 2023. URL https://arxiv.org/abs/2304.03442.

Noah Shinn, Federico Cassano, Edward Berman, et al. Reflexion: Language agents with verbal reinforcement learning. In *NeurIPS*, 2023.

Linrui Tian, Qi Wang, Bang Zhang, and Liefeng Bo. Emo: Emote portrait alive generating expressive portrait videos with audio2video diffusion model under weak conditions. In *ECCV*, pp. 244–260. Springer, 2024.

Linrui Tian, Siqi Hu, Qi Wang, Bang Zhang, and Liefeng Bo. Emo2: End-effector guided audio-driven avatar video generation. *arXiv preprint arXiv:2501.10687*, 2025.

Shuyuan Tu, Yueming Pan, Yinming Huang, Xintong Han, Zhen Xing, Qi Dai, Chong Luo, Zuxuan Wu, and Jiang Yu-Gang. Stableavatar: Infinite-length audio-driven avatar video generation. *arXiv preprint arXiv:2508.08248*, 2025.

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

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

Guanzhi Wang, Yuqi Xie, Yunfan Jiang, et al. Voyager: An open-ended embodied agent with large language models. *Transactions on Machine Learning Research*, 2024a.

Kangrui Wang, Pingyue Zhang, Zihan Wang, Yaning Gao, Linjie Li, Qineng Wang, Hanyang Chen, Chi Wan, Yiping Lu, Zhengyuan Yang, et al. Vagen: Reinforcing world model reasoning for multi-turn vlm agents. In *NeurIPS*, 2025a.

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

Zihan Wang, Kangrui Wang, Qineng Wang, Pingyue Zhang, Linjie Li, Zhengyuan Yang, Xing Jin, Kefan Yu, Minh Nhat Nguyen, Licheng Liu, et al. Ragen: Understanding self-evolution in llm agents via multi-turn reinforcement learning. *arXiv preprint arXiv:2504.20073*, 2025b.

Zora Zhiruo Wang, Jiayuan Mao, Daniel Fried, and Graham Neubig. Agent workflow memory. In *Forty-second International Conference on Machine Learning*, 2025c. URL https://openreview.net/forum?id=NTAhi2JEEE.

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.

Jiazheng Xu, Yu Huang, Jiale Cheng, Yuanming Yang, Jiajun Xu, Yuan Wang, Wenbo Duan, Shen Yang, Qunlin Jin, Shurun Li, Jiayan Teng, Zhuoyi Yang, Wendi Zheng, Xiao Liu, Dan Zhang, Ming Ding, Xiaohan Zhang, Xiaotao Gu, Shiyu Huang, Minlie Huang, Jie Tang, and Yuxiao Dong. Visionreward: Fine-grained multi-dimensional human preference learning for image and video generation. In *AAAI*, 2026.

Mingwang Xu, Hui Li, Qingkun Su, Hanlin Shang, Liwei Zhang, Ce Liu, Jingdong Wang, Yao Yao, and Siyu Zhu. Hallo: Hierarchical audio-driven visual synthesis for portrait image animation. *arXiv preprint arXiv:2406.08801*, 2024.

Wujiang Xu, Kai Mei, Hang Gao, Juntao Tan, Zujie Liang, and Yongfeng Zhang. A-mem: Agentic memory for llm agents. *Advances in Neural Information Processing Systems*, 2025.

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

Shaoshu Yang, Zhe Kong, Feng Gao, Meng Cheng, Xiangyu Liu, Yong Zhang, Zhuoliang Kang, Wenhan Luo, Xunliang Cai, Ran He, and Xiaoming Wei. Infinitetalk: Audio-driven video generation for sparse-frame video dubbing. *ArXiv*, abs/2508.14033, 2025b. URL https://api.semanticscholar.org/CorpusID:280686279.

Zhendong Yang, Ailing Zeng, Chun Yuan, and Yu Li. Effective whole-body pose estimation with two-stages distillation. In *ICCV*, 2023.

Zhuoyi Yang, Jiayan Teng, Wendi Zheng, Ming Ding, Shiyu Huang, Jiazheng Xu, Yuanming Yang, Wenyi Hong, Xiaohan Zhang, Guanyu Feng, et al. Cogvideox: Text-to-video diffusion models with an expert transformer. In *ICLR*, 2025c.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. React: Synergizing reasoning and acting in language models. In *ICLR*, 2023.

Shenghai Yuan, Yuanyang Yin, Zongjian Li, Xinwei Huang, Xiao Yang, and Li Yuan. Helios: Real real-time long video generation model. *arXiv preprint arXiv:2603.04379*, 2026.

Zhengqing Yuan, Yixin Liu, Yihan Cao, Weixiang Sun, Haolong Jia, Ruoxi Chen, Zhaoxu Li, Bin Lin, Li Yuan, Lifang He, et al. Mora: Enabling generalist video generation via a multi-agent framework. *arXiv preprint arXiv:2403.13248*, 2024.

Mert Yuksekgonul, Federico Bianchi, Joseph Boen, Sheng Liu, Zhi Huang, Carlos Guestrin, and James Zou. Textgrad: Automatic "differentiation" via text. *Nature*, 2025.

Jiacheng Zhang, Jie Wu, Weifeng Chen, Yatai Ji, Xuefeng Xiao, Weilin Huang, and Kai Han. Onlinevpo: Align video diffusion model with online video-centric preference optimization, 2025a. URL https://arxiv.org/abs/2412.15159.

Qizheng Zhang, Changran Hu, Shubhangi Upasani, Boyuan Ma, Fenglu Hong, Vamsidhar Kamanuru, Jay Rainton, Chen Wu, Mengmeng Ji, Hanchen Li, Urmish Thakker, James Zou, and Kunle Olukotun. Agentic context engineering: Evolving contexts for self-improving language models. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=eC4ygDs02R.

Wenxuan Zhang, Xiaodong Cun, Xuan Wang, Yong Zhang, Xi Shen, Yu Guo, Ying Shan, and Fei Wang. Sadtalker: Learning realistic 3d motion coefficients for stylized audio-driven single image talking face animation. In *CVPR*, 2023.

Zhihong Zhang, Xiaojian Huang, Jin Xu, Zhuodong Luo, Xinzhi Wang, Jiansheng Wei, and Xuejin Chen. Videorewardbench: Comprehensive evaluation of multimodal reward models for video understanding, 2025b. URL https://arxiv.org/abs/2509.00484.

Zhimeng Zhang, Lincheng Li, Yu Ding, and Changjie Fan. Flow-guided one-shot talking face generation with a high-resolution audio-visual dataset. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3661–3670, 2021.

Andrew Zhao, Daniel Huang, Quentin Xu, Matthieu Lin, Yong-Jin Liu, and Gao Huang. Expel: Llm agents are experiential learners. In *AAAI*, 2024.

Dingcheng Zhen, Xu Zheng, Ruixin Zhang, Zhiqi Jiang, Yichao Yan, Ming Tao, and Shunshun Yin. Soulx-liveact: Towards hour-scale real-time human animation with neighbor forcing and convkv memory, 2026. URL https://arxiv.org/abs/2603.11746.

Dian Zheng, Ziqi Huang, Hongbo Liu, Kai Zou, Yinan He, Fan Zhang, Lulu Gu, Yuanhan Zhang, Jingwen He, Wei-Shi Zheng, Yu Qiao, and Ziwei Liu. VBench-2.0: Advancing video generation benchmark suite for intrinsic faithfulness, 2025. URL https://arxiv.org/abs/2503.21755.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric P. Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. Judging llm-as-a-judge with mt-bench and chatbot arena. In *NeurIPS Datasets and Benchmarks Track*, 2023.

Longtao Zheng, Rundong Wang, Xinrun Wang, and Bo An. Synapse: Trajectory-as-exemplar prompting with memory for computer control. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=Pc8AU1aF5e.

Yong Zhong, Min Zhao, Zebin You, Xiaofeng Yu, Changwang Zhang, and Chongxuan Li. Posecrafter: One-shot personalized video synthesis following flexible pose control. In *ECCV*, 2024.

Bingwen Zhu, Fanyi Wang, Tianyi Lu, Peng Liu, Jingwen Su, Jinxiu Liu, Yanhao Zhang, Zuxuan Wu, Yu-Gang Jiang, and Guo-Jun Qi. Poseanimate: Zero-shot high fidelity pose controllable character animation. *arXiv*, 2024a.

Shenhao Zhu, Junming Leo Chen, Zuozhuo Dai, Yinghui Xu, Xun Cao, Yao Yao, Hao Zhu, and Siyu Zhu. Champ: Controllable and consistent human image animation with 3d parametric guidance. In *ECCV*, 2024b.

A TECHNICAL APPENDIX

APPENDIX CONTENTS

A.1 Dataset Details 16
A.2 Self-Evolving Algorithm 17
A.3 Planner System-Prompt Structure 18
A.4 Critique Agent Evaluation Procedure 19
A.5 Correlation with Human Judgments 20
A.6 Structured Skill Library Details 22
A.7 Real-Time Interactive Avatar Framework 24
A.8 Limitations 24
A.9 Future Work 25

A.1 DATASET DETAILS

As described in Section 4.1, our benchmark comprises two families of avatar reference images: a half-body tabletop family of ten desk-scale scenes and a full-body family of ten scenes with a larger motion space. Figure 4 shows representative task instances from both families. Each goal is VLM-generated and human-filtered so that it decomposes into 3–6 semantically explicit operation steps, and the goal description explicitly lists the visible task objects to ground subsequent action prompting. The framing tag (half-body or full-body) follows the reference image. We also evaluate zero-shot transfer on 100 L-IVA cases (5 scenes $\times$ 20) (He et al., 2025b).

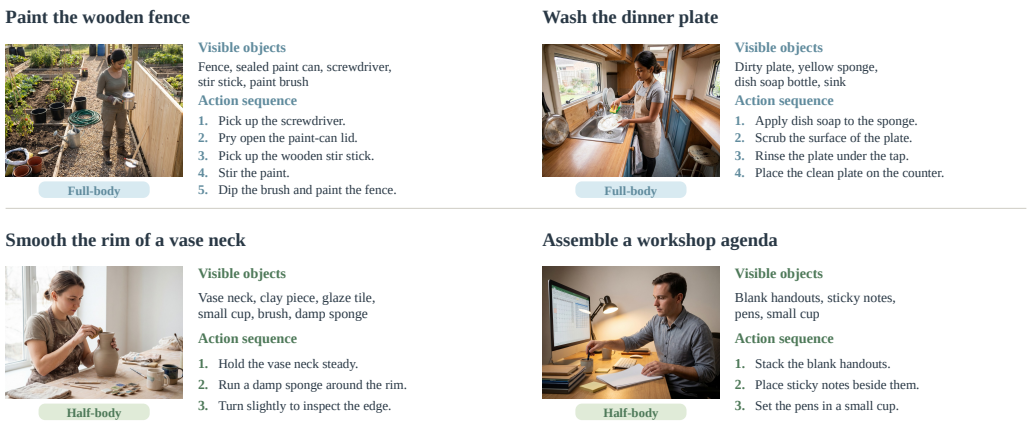


Figure 4: Representative task instances: full-body scenes (top) and half-body tabletop scenes (bottom). Each panel summarizes visible objects and action steps; complete task descriptions are provided below.

Complete task descriptions. The four original task prompts are reproduced below in row-wise order, with one duplicated article corrected for readability.

Paint the wooden fence (top left, full-body). The person must paint the wooden fence, using visible items such as unpainted wooden fence, sealed paint can, screwdriver, wooden stir stick, paint brush. The action should unfold as a clear multi-step sequence: pick up the screwdriver; pry open the paint can lid; pick up the stir stick; stir the paint; dip the brush and paint the fence. This requires planning, sequencing, and careful object manipulation while staying grounded in a realistic garden environment.

Wash the dirty dinner plate (top right, full-body). The person must wash the dirty dinner plate, using visible items such as dirty dinner plate, yellow sponge, dish soap bottle, sink. The action should unfold as a clear multi-step sequence: apply dish soap to the sponge; scrub the surface of the plate; rinse the plate under the tap; place the clean plate on the counter. This requires planning, sequencing, and careful object manipulation while staying grounded in a realistic kitchen environment.

Smooth Rim of a vase neck (bottom left, half-body). The person should smooth the rim of a vase neck. The visible sequence is: hold the vase neck steady; then run a damp sponge around the rim; then turn it slightly to inspect the edge. The scene should be a realistic half-body tabletop interaction with clear hands, clear face, and enough object detail for later action prompting. Visible task objects include a vase neck, clay piece, glaze tile, small cup, brush, sponge.

Assemble a Workshop Agenda (bottom right, half-body). The person should assemble agenda materials for a workshop. The visible sequence is: stack blank handouts; then place sticky notes beside them; then set pens in a small cup. The scene should be a realistic half-body tabletop interaction with clear hands, clear face, and enough object detail for later action prompting.

A.2 Self-Evolving Algorithm

Algorithm 1 Self-Evolution for Video Avatars

Require: Evolution task set $\mathcal{D}_{\text{evo}}$; test goal (I_{ref}, g); initial context $\mathcal{C}^{(0)} = (P_{\text{sys}}^{(0)}, \mathcal{S}^{(0)})$; frozen generator G; VLM critique agent E; update agent R; evolution strategy $\sigma \in \{\text{HCE}, \text{CCE}\}$; number of rounds N
Ensure: Generated video $V^\star$; selected context $\mathcal{C}^\star = (P_{\text{sys}}^\star, \mathcal{S}^\star)$

for $t = 0$ to $N-1$ **do**
  $\mathcal{T}^{(t)} \Leftarrow \text{CollectTrajectories}(\mathcal{D}_{\text{evo}}, \mathcal{C}^{(t)}, \sigma)$ {plan, render, critique}
  $\mathcal{C}^{(t+1)} \Leftarrow \text{StrategyUpdate}(\mathcal{C}^{(t)}, \mathcal{T}^{(t)}, \sigma)$ {context-only update}
end for
$\mathcal{C}^\star \Leftarrow \text{SelectCheckpoint}(\{\mathcal{C}^{(t)}\}_{t=0}^{N})$ {freeze the specified checkpoint}
$\mathcal{S}_{\text{rel}}^\star \Leftarrow \varnothing$
if skill retrieval is enabled **then**
  $\mathcal{S}_{\text{rel}}^\star \Leftarrow \text{Retrieve}(\mathcal{S}^\star, I_{\text{ref}}, g)$
end if
$\text{Plan}^\star \Leftarrow \pi(I_{\text{ref}}, g; P_{\text{sys}}^\star, \mathcal{S}_{\text{rel}}^\star)$
$V^\star \Leftarrow \text{Render}(G, I_{\text{ref}}, \text{Plan}^\star)$ {single planning pass with segment-wise generation}
return $V^\star$, $P_{\text{sys}}^\star$, $\mathcal{S}^\star$

In Algorithm 1, SelectCheckpoint loads the checkpoint specified by the evaluation configuration before held-out testing; the checkpoint remains fixed throughout evaluation. A trajectory records, for one goal $x_i = (I_i, g_i)$, the sampled plan(s), the video rendered segment by segment through the frozen backend G, and the critique feedback. For each non-final segment, $\text{SelectAnchor}(I_i, \text{LastFrame}(v_j), p_{j+1})$ chooses the next conditioning frame according to the plan's continuity declaration: either the preceding segment's last frame or the original reference image I_i. The strategies in Section 3.2 share the outer loop and differ in trajectory collection and context revision. **HCE** samples one plan per goal and, after evaluating all goals, applies one global revision, $\mathcal{C}^{(t+1)} = R_{\text{hol}}(\mathcal{C}^{(t)}, \{(p_i, f_i)\}_i)$, for N=12 rounds. **CCE** samples L=4 plans per goal, $p_{i,1:L} \sim \pi(x_i; \mathcal{C}^{(t)})$. Let $s_{i,\ell}$ denote the SBM failure count of rendered video $V_{i,\ell}$, evaluated against x_i and $p_{i,\ell}$. CCE selects $\ell_i^+ = \arg\min_\ell s_{i,\ell}$ and $\ell_i^- = \arg\max_\ell s_{i,\ell}$, takes the corresponding plans and feedback, and merges per-goal edits $\Delta_i = R_{\text{con}}(\mathcal{C}^{(t)}, p_i^+, p_i^-, f_i^+, f_i^-)$; the component ablation uses N=6 rounds, while Table 1 retains this six-round result alongside the exact N=3 prompt evaluated with skill retrieval disabled. Both updaters receive plans and critique feedback and use prompt-guided reasoning to propose context edits, whereas SBM counts all observed semantic failures. All model parameters remain frozen; only $\mathcal{C}^{(t)}$ evolves. The per-round procedures appear in Algorithms 2 and 3.

CCE evaluation records. Table 1 preserves two evaluations on 400 held-out cases (200 per framing). The N=3 row uses the exact v3 prompt, with skill retrieval disabled, one test-time plan, and

complete-video inputs to all five critics. The $N{=}6$ row retains the earlier checkpoint selected from a six-round evolution run; it does not imply that every scene uses the exact v6 prompt. In that earlier evaluation, 369 critiques used complete videos and 31 used sampled frames. Those historical scores are retained without rescoring. The third-round renders use random seeds that are not matched to the historical runs, so the two rows record distinct evaluation settings rather than isolating the causal effect of additional rounds.

Algorithm 2 Single-plan Holistic Context Evolution (HCE): One Round

Require: Evolution task set $\mathcal{D}_{\text{evo}}$; context $\mathcal{C}^{(t)} = (P_{\text{sys}}^{(t)}, \mathcal{S}^{(t)})$; frozen generator G; VLM critique agent E; update agent R
Ensure: Updated context $\mathcal{C}^{(t+1)}$
$\mathcal{T} \Leftarrow [\,]$
for each $x_i = (I_i, g_i) \in \mathcal{D}_{\text{evo}}$ **do**
  $\mathcal{S}_{\text{rel}} \Leftarrow \text{Retrieve}(\mathcal{S}^{(t)}, I_i, g_i)$
  $p_i \Leftarrow \pi(I_i, g_i; P_{\text{sys}}^{(t)}, \mathcal{S}_{\text{rel}})$ {one plan per goal}
  $V_i \Leftarrow [\,]$, $a_1 \Leftarrow I_i$
  for each segment $p_j \in p_i$ **do**
    $v_j \Leftarrow G(a_j, p_j)$; $V_i.\text{append}(v_j)$
    if $j < |p_i|$ **then**
      $a_{j+1} \Leftarrow \text{SelectAnchor}(I_i, \text{LastFrame}(v_j), p_{j+1})$
    end if
  end for
  $f_i \Leftarrow E(I_i, g_i, p_i, V_i)$
  $\mathcal{T}.\text{append}((p_i, f_i))$
end for
$\mathcal{C}^{(t+1)} \Leftarrow R_{\text{hol}}(\mathcal{C}^{(t)}, \{(p_i, f_i)\}_i)$ {prompt-guided revision}
return $\mathcal{C}^{(t+1)}$

Algorithm 3 Multi-plan Contrastive Context Evolution (CCE): One Round

Require: Evolution task set $\mathcal{D}_{\text{evo}}$; context $\mathcal{C}^{(t)} = (P_{\text{sys}}^{(t)}, \mathcal{S}^{(t)})$; frozen generator G; VLM critique agent E; update agent R; sample count $L = 4$
Ensure: Updated context $\mathcal{C}^{(t+1)}$
for each $x_i = (I_i, g_i) \in \mathcal{D}_{\text{evo}}$ **do**
  $\mathcal{S}_{\text{rel}} \Leftarrow \text{Retrieve}(\mathcal{S}^{(t)}, I_i, g_i)$
  for $\ell = 1$ to L **do**
    $p_{i,\ell} \sim \pi(I_i, g_i; P_{\text{sys}}^{(t)}, \mathcal{S}_{\text{rel}})$ {sampled plans}
    $V_{i,\ell} \Leftarrow \text{Render}(G, I_i, p_{i,\ell})$ {segment-wise with anchor selection}
    $f_{i,\ell} \Leftarrow E(I_i, g_i, p_{i,\ell}, V_{i,\ell})$
    $s_{i,\ell} \Leftarrow \text{SBM}(x_i, p_{i,\ell}, V_{i,\ell})$ {all observed semantic failures}
  end for
  $\ell_i^+ \Leftarrow \arg\min_\ell s_{i,\ell}$, $\ell_i^- \Leftarrow \arg\max_\ell s_{i,\ell}$
  $(p_i^\pm, f_i^\pm) \Leftarrow (p_{i,\ell_i^\pm}, f_{i,\ell_i^\pm})$ {selected plans and feedback}
  $\Delta_i \Leftarrow R_{\text{con}}(\mathcal{C}^{(t)}, p_i^+, p_i^-, f_i^+, f_i^-)$ {prompt-guided revision}
end for
$\mathcal{C}^{(t+1)} \Leftarrow \text{Merge}(\mathcal{C}^{(t)}, \{\Delta_i\}_i)$ {merge per-goal edits}
return $\mathcal{C}^{(t+1)}$

A.3 Planner System-Prompt Structure

The planner is controlled by the system prompt P_{sys}. The base prompt comprises the following parts:

1. **Role definition**: constrains the model to an "avatar behavior video planner," emphasizing goal completion and scene consistency;
2. **Output format**: prescribes a JSON-structured output covering scene analysis, task decomposition, and segment descriptions;
3. **Planning rules**: limit the number of actions per segment, require explicit declaration of object-state changes, and maintain camera and human-pose continuity;

4. **Skill-injection slot**: dynamically inserts relevant skills retrieved from the skill library.

A.4 Critique Agent Evaluation Procedure

The complete-video critique procedure, used for the exact $N{=}3$ CCE evaluation, operates as follows:

1. assemble the reference image, interaction goal, segment plan, and complete rendered video into multimodal inputs;
2. submit the complete video to each of five specialized VLM critics, one per SBM dimension, without replacing it with sampled frames;
3. collect structured failure records with a dimension, description, and localization information, and validate each critic's response and video input;
4. deduplicate the records within each dimension and video, then sum the five counts according to Equation (3).

In this evaluation, failed or invalid critiques are retried with complete-video inputs; sampled-frame and text-only fallbacks are not accepted. The historical six-round CCE record includes 31 sampled-frame critiques, as documented separately in Appendix A.2. VMBench uses a separate preprocessing path: for the third-round evaluation, each video is uniformly resampled to 49 frames and encoded at 8 fps before visual-quality scoring. This resampling does not affect the video submitted to the SBM critics.

For SBM aggregation, descriptions are normalized by stripping leading and trailing whitespace and converting to lowercase. Records with the same dimension and normalized description are counted once within a video, regardless of their segment indices or timestamps. Distinct descriptions remain separate records, and a failure reported under different dimensions contributes to each dimension. There is no additional semantic deduplication or normalization by the number of segments; reported results are mean failure counts per task.

The five dimensions capture complementary failure sources: PPF checks physical plausibility of the visible action; OIF checks that manipulated objects preserve identity, geometry, and state; and CTF checks temporal and state consistency across adjacent clips. GCF jointly considers the goal, the plan, and the complete sequence, capturing both omitted steps and unsuccessful execution, while CMF assesses commonsense consistency and flags violations arising from either the plan or its execution.

The critic prompt asks for semantic and physical failures rather than purely aesthetic defects. These observations are scored independently of their causal origin: for example, a disappearing pen can count as a physical failure without establishing that the plan caused it. During evolution, the updater receives the plans and critique records and is prompted to infer reusable changes to action decomposition, object-state tracking, and continuity, rather than optimize rendering style or camera aesthetics. Thus an underspecified placement destination can motivate an explicit destination rule, whereas an isolated visual defect alone does not justify that rule. This attribution is a prompt-guided judgment during revision; the implementation supplies no separate binary attribution field or hard filtering step. It aims to reduce noise-driven context changes without guaranteeing correct causal attribution.

Listing 1 shows an example output of a single specialized critic agent (PPF dimension): each failed case localizes one visible failure to a segment index and a timestamp range, together with a concrete description of what is observed.

Listing 1: Example output of one specialized critic agent (PPF dimension) for a generated video. Each failed case carries the SBM metric, the failure category, the related segment index, a timestamp range, and a concrete description of the visible failure.


```
{
  "verdict": "needs_improvement",
  "failed_cases": [
    {
      "metric": "ppf",
      "category": "physical plausibility",
      "related_segment_index": 1,
      "timestamp_start_sec": 0.0,
      "timestamp_end_sec": 6.0,
      "description": "The dark pen held in the person's right hand gradually dissolves and
     disappears into the hand and the paper beneath it."
    },
    {
      "metric": "ppf",
      "category": "physical plausibility",
      "related_segment_index": 2,
      "timestamp_start_sec": 7.0,
      "timestamp_end_sec": 10.0,
      "description": "The fingers of the right hand resting on the chart are anatomically
     malformed, appearing mangled and fused together."
    },
    {
      "metric": "ppf",
      "category": "physical plausibility",
      "related_segment_index": 4,
      "timestamp_start_sec": 7.0,
      "timestamp_end_sec": 19.0,
      "description": "The pen held in the left hand continuously morphs in thickness, and
     the fingers gripping it melt and fuse into the body of the pen, creating an anatomically
      impossible grip."
    }
  ]
}
```


A.5 Correlation with Human Judgments

Study protocol. We conduct a human study following the L-IVA evaluation protocol (He et al., 2025b). Eight human evaluators independently annotate all 100 videos randomly sampled from the L-IVA evaluation outputs. All evaluators assess every video. For each video, they are presented with the high-level intention, the complete reference-subgoal list, and the generated video sequence. Videos are anonymized and displayed in randomized order to prevent method-identity and presentation-order biases. Although two videos are displayed side by side for annotation efficiency, each receives an independent PPS rating and independent subgoal-completion labels.

Physical Plausibility Score (PPS). PPS evaluates physical consistency across the full video sequence, focusing on gravity, collision, contact, object permanence, spatial consistency, and physically implausible hallucinations. Each evaluator assigns an integer score on a five-point Likert scale: 5 (*Perfect*) indicates realistic physical interactions with correct gravity, collision, and contact; 4 (*Good*) indicates generally correct physics with only minor artifacts that do not impede comprehension; 3 (*Fair*) indicates noticeable floating, clipping, or related artifacts while the action logic remains coherent; 2 (*Poor*) indicates severe physical violations such as object teleportation or interpenetration; and 1 (*Fail*) indicates a complete breakdown of basic physical consistency.

Task Success Rate (TSR). TSR measures whether the generated sequence fulfills the high-level intention through successful completion of its constituent subgoals. For each video, an evaluator examines every reference subgoal and assigns a binary completion label based on visible evidence. A subgoal is counted as completed only when its intended action and resulting state transition are visibly realized with sufficient physical plausibility; merely attempting the action, showing an ambiguous intermediate state, or producing the target appearance through an obvious physical violation does not count as successful completion. For video i with M_i reference subgoals, the evaluator-level

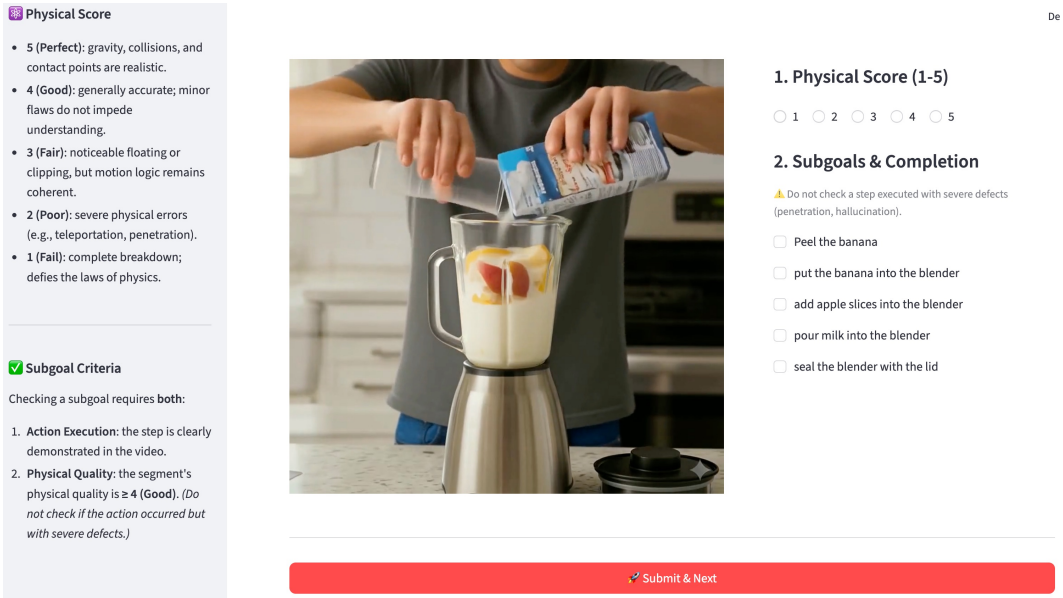


Figure 5: Interface of the blind side-by-side human-rating study. For each case, two anonymized videos are presented in randomized order, while each video is evaluated independently. The interface displays the generated video together with the five-point Physical Plausibility Score (PPS) rubric on the left and the reference-subgoal checklist used to compute Task Success Rate (TSR) on the right.

TSR assigned by evaluator r is

$$\mathrm{TSR}_i^{(r)} = \frac{1}{M_i} \sum_{m=1}^{M_i} y_{i,m}^{(r)},$$

where $y_{i,m}^{(r)} \in \{0, 1\}$ indicates whether evaluator r judges the m-th subgoal as completed.

Human-score aggregation. For each video and each human-evaluation metric, we first obtain eight evaluator-level scores. To reduce the influence of isolated extreme judgments, we remove one highest and one lowest value and average the remaining six values. Specifically, for metric $h \in \{\mathrm{PPS}, \mathrm{TSR}\}$, the aggregated video-level score is

$$\bar{h}_i = \frac{1}{6} \left(\sum_{r=1}^{8} h_i^{(r)} - \max_r h_i^{(r)} - \min_r h_i^{(r)} \right).$$

We compute Spearman's rank correlations between aggregated human scores and SBM failure metrics over the 100 videos. Individual evaluator ratings are not treated as independent observations.

Table 5: Spearman rank correlations between SBM failure metrics and aggregated human judgments on 100 randomly sampled L-IVA videos. All eight evaluators assess every video. For each video and human-evaluation metric, one highest and one lowest evaluator-level value are discarded, and the remaining six values are averaged before computing correlations. Lower SBM failure counts indicate better performance, whereas higher PPS and TSR values indicate better performance; therefore, semantic alignment is reflected by negative correlations.

SBM metric (↓)	PPS (↑)	TSR (↑)
TPF	−0.71	−0.48
PPF	**−0.79**	−0.30
CMF	−0.28	−0.53
GCF	−0.31	**−0.63**
CTF	−0.59	−0.27
OIF	−0.45	−0.07

Correlation analysis. The observed correlation pattern is consistent with the intended semantics of the SBM dimensions. PPF shows the strongest observed association with human-rated physical plausibility ($\rho = -0.79$), supporting its role in detecting physically implausible actions and interactions. CTF is also substantially associated with PPS ($\rho = -0.59$), indicating that discontinuities in object states, poses, and spatial configurations across adjacent clips contribute to human perceptions of physical implausibility. GCF shows the strongest observed association with human-evaluated task success ($\rho = -0.63$), followed by CMF ($\rho = -0.53$), consistent with their focus on goal completion and commonsense consistency. The aggregate TPF score is associated with both PPS ($\rho = -0.71$) and TSR ($\rho = -0.48$), suggesting that it captures broader execution quality across multiple semantic failure sources. In contrast, OIF has only a weak association with TSR ($\rho = -0.07$), showing that preserving object identity and geometry is related to, but not equivalent to, completing the intended task. Overall, the dimension-aligned correlation pattern provides evidence that SBM captures meaningful aspects of both physical execution quality and long-horizon task success while retaining distinct evaluation focuses across its subdimensions.

A.6 Structured Skill Library Details

Skill representation. The skill library $\mathcal{S}$ stores transferable experience abstracted from historical tasks. Each skill is represented as a structured JSON record containing:

- skill_id: a unique kebab-case identifier for the skill;
- title: a short human-readable name;
- scene_tags: applicable scene tags used for retrieval;
- task_pattern: a regular expression matching the tasks the skill applies to;
- planning_rules: the concrete rules injected into the planner;
- anti_patterns: the erroneous patterns to avoid.

Runtime records additionally carry bookkeeping fields (source iteration, confidence, and version), which are omitted from the listing. Listing 2 shows three representative skills learned for the pottery scene, including the wedge-clay skill illustrated in the running example of Figure 2.

Skill learning. The skill-learning module extracts transferable patterns from critique agent feedback and planning text (Anthropic, 2026; Zhao et al., 2024). For recurring failures, the system analyzes the corresponding planning defect and abstracts the repair rule into a skill; for consistently successful cases, it extracts the underlying planning pattern as a positive template. Structured skills are shorter, more controllable, and better suited for cross-scene injection than raw trajectories.

Skill retrieval and injection. At inference, the system first loads all global skills, then retrieves scene-specific skills according to the scene type and task keywords, and inserts them into the skill region of the system prompt. The three most stable general skills observed in experiments are *global object-state tracking*, *camera and pose anchoring*, and *single-physical-transition constraint*; these recur across scenes because they address largely scene-independent failure modes, including loss of object integrity, unstable viewpoints, and segments containing too many actions.

Listing 2: Three representative learned skills from the pottery scene. Each record pairs concrete planning rules with the anti-patterns they suppress; bookkeeping fields (source iteration, confidence, version) are omitted for layout.


```
[
  {
    "skill_id": "wedge-clay",
    "scene_tags": ["clay", "wedging", "pottery"],
    "task_pattern": "Wedge .* clay",
    "title": "Wedge Clay",
    "planning_rules": [
      "Describe clay as a thick, dense, cohesive solid mass that maintains its volume. It
     must not tear, stretch into a tall pillar, flatten into a thin sheet, or spontaneously
     generate liquid.",
      "The hands must firmly grip the sides of the clay mass. Do not describe the hands as
     sliding over the surface or pulling the clay upwards.",
      "For folding, DO NOT use words like 'pull', 'stretch', or 'lift high'. Explicitly
     state the hands press down and roll the top of the clay toward the body, keeping the
     clay as a low, compact mound on the table.",
      "For placement, explicitly state that the hands grasp the clay, carry it to the target
      location, and set it flat on the empty table surface *next to* the damp sponge, leaving
      a visible gap. Do not place it on top of the sponge.",
      "Maintain a static, stable half-body camera framing throughout all segments.
     Absolutely no zooming in, changing angles, or cropping the person's face."
    ],
    "anti_patterns": [
      "Clay tearing, stretching vertically, dripping liquid, or flattening into a thin sheet
     .",
      "Hands sliding over the clay or pulling it upwards.",
      "Placing the clay on top of the sponge or moving hands away empty.",
      "Camera zooming in, changing angles, or cropping the person's face."
    ]
  },
  {
    "skill_id": "compare-and-choose-glaze",
    "scene_tags": ["glaze tiles", "comparison", "pottery"],
    "task_pattern": "Choose Glaze for .*",
    "title": "Compare and Choose Glaze",
    "planning_rules": [
      "Glaze tiles are flat, rigid, dry ceramic squares. They must NOT transform into
     brushes, sponges, or vessels, and must NOT be used to paint or pour liquid.",
      "When picking up a tile, explicitly state that the fingers pinch the outer edges of
     the tile. Do not place the palm flat on top of the tile.",
      "When comparing, explicitly state that the person's hands hold the two tiles low,
     hovering just above the table, one on the left side and one on the right side of the
     target object's base. They must be held *beside* the object, not high near the face, and
      must *not touch* it.",
      "For the final placement, explicitly describe moving *only the chosen tile* forward
     and setting it flat on the *empty table surface* directly in front of the target object.
      It must NOT be placed on top of the target object. The unchosen tile must remain held
     up in the other hand.",
      "Maintain a static, stable half-body camera framing throughout all segments.
     Absolutely no zooming in, changing angles, or cropping the person's face."
    ],
    "anti_patterns": [
      "Glaze tiles morphing into brushes/sponges, fusing together, or pouring liquid.",
      "Using a glaze tile to paint the cup or table.",
      "Holding tiles high near the face or directly above the target object.",
      "Placing the chosen tile on top of the target object instead of the empty table.",
      "Camera zooming in, changing angles, or cropping the person's face."
    ]
  }
]
```

A.7 Real-Time Interactive Avatar Framework

We evaluate Helios (Yuan et al., 2026) on a test set of 40 videos randomly sampled from HDTF (Zhang et al., 2021), and compare it with SoulX-LiveAct (Zhen et al., 2026) and LiveAvatar (Huang et al., 2025). We report audio–visual synchronization using Sync-C and Sync-D, together with subject consistency, background consistency, motion smoothness, dynamic degree, aesthetic quality, and imaging quality. Higher values are better except for Sync-D.

Table 6: Audio-driven video-generation results on a test set of 40 videos randomly sampled from HDTF. Sync-C and Sync-D measure audio–visual synchronization, while the remaining metrics measure perceptual video quality. Higher values indicate better performance for all metrics except Sync-D. The best result for each metric is shown in bold.

Metric	**Helios (ours)**	SoulX-LiveAct	LiveAvatar
Sync-C ↑	**7.44**	6.55	7.08
Sync-D ↓	8.40	9.10	**8.27**
Subject consistency ↑	0.829	0.973	**0.982**
Background consistency ↑	0.860	0.955	**0.965**
Motion smoothness ↑	0.987	0.991	**0.992**
Dynamic degree ↑	**1.000**	**1.000**	0.975
Aesthetic quality ↑	0.448	**0.529**	0.522
Imaging quality ↑	0.407	**0.670**	0.649

Streaming efficiency. We further compare the time to first chunk (TTFC), steady-state generation latency per chunk, and streaming real-time factor (RTF). Before measurement, each method is warmed up to exclude initialization and model-loading overhead. All latency and RTF values are reported as averages over 40 runs. TTFC measures the delay before the first generated video chunk becomes available, while steady-state latency measures the generation time of subsequent chunks. The streaming RTF is defined as the ratio of generation time to generated video duration, where an RTF below 1 indicates faster-than-real-time generation.

Table 7: Streaming-latency comparison after model warm-up. All reported latency and RTF values are averaged over 40 runs. For SoulX-LiveAct, 1.2 s and 1.6 s denote the first and subsequent chunk durations, respectively. TTFC denotes the time to the first generated chunk, steady/chunk denotes the generation latency of subsequent chunks, and RTF denotes the ratio of generation time to generated video duration. Values marked with ∼ are approximate. Helios achieves the lowest average steady-state latency and streaming RTF among the compared methods.

Method	Chunk duration	TTFC ↓	Steady/chunk ↓	RTF ↓
Helios (ours)	1.32 s	**0.78** s	**0.77** s	**0.60**
SoulX-LiveAct	1.2/1.6 s	∼0.80 s	∼1.19 s	∼0.74
LiveAvatar	∼1.9 s	24–42 s	∼35.5 s	∼19.0

After model warm-up, Helios generates a 1.32 s video chunk in an average of 0.77 s during steady-state streaming over 40 runs, corresponding to an average RTF of 0.60 on a single NVIDIA H100 GPU. Its average TTFC is 0.78 s. These results demonstrate that Helios supports low-latency, faster-than-real-time audio-conditioned avatar generation.

A.8 Limitations

VAEM has several limitations. First, because the video-generation backend is kept frozen to isolate the effect of context evolution, the achievable behavior remains bounded by the capability of the underlying renderer. Planning improvements cannot fully correct severe execution failures caused by large motions, complex object manipulation, or long-horizon generation. Second, SBM relies on VLM-based critics and may inherit evaluator-specific biases despite its fine-grained and interpretable design. Finally, although prompt rules and retrieved skills provide complementary benefits,

their combination is not uniformly additive across all evaluation dimensions, suggesting occasional interference between global behavioral constraints and task-specific procedural knowledge.

A.9 Future Work

Future work will extend VAEM beyond task execution toward long-term human–avatar interaction. In particular, we aim to equip avatars with persistent user-specific memory and consistent personas, allowing them to accumulate interaction histories, adapt their behavior through self-evolution, and support real-time, sustained emotional companionship. For evaluation, we plan to further calibrate SBM with larger-scale human annotations, introduce evaluator models independent of the critics used for context evolution, and develop normalized metrics for plans of different lengths. Finally, reducing the cost of offline context evolution and enabling safe continual online updates will be important for deploying self-evolving video avatars in long-running interactive applications.