Compile the Event Logic Before Rendering

Executable temporal constraints for multi-round Agentic Vision

Abstract

Text-to-video models can produce individually plausible actions while violating the order, preconditions, or exclusions that make a procedure coherent. We propose an Agentic Vision architecture in which one multimodal controller reads instructions, writes an executable event plan, inspects Blender previews and generated video, and repairs localized violations across multiple rounds. We will test whether explicit event constraints survive neural rendering and improve all-constraint satisfaction at matched generation cost.

Keywords / agentic vision, text-to-video, temporal logic, Blender, visual verification

Problem and prior work

Consider a demonstration in which a lid must close before a container tilts, and a second container must remain untouched. A global prompt score may reward the presence of both actions while missing their relation. VideoDirectorGPT [1] uses LLM planning to generate grounded scenes, and VideoCoCo [2] already uses executable Blender code to mediate physical video generation. Our proposed contribution is to verify explicit, compositional event relations against both executable scene state and final pixels, then test whether this dual check adds value beyond iterative rendering.

Unified understanding and generation loop

A shared multimodal controller interprets the prompt, inspects frames, and selects edits using one typed scene record. Compile a bounded vocabulary of events into Python predicates over stable object IDs and frame intervals. Support precedence, overlap, preconditions, and forbidden events. A Blender script schedules animation and emits a preview, object masks, and depth. A frozen control-capable editor, such as VACE [3], converts control renders into the final appearance. The controller can call crop, frame seek, segmentation and tracking tools to obtain time-stamped evidence from the final frames. Keep the symbolic plan, scene code and visual evidence linked by object ID and interval.

Multi-round repair

First repair infeasible schedules in code. After generation, a failed predicate returns a suspected event, objects, and interval. The controller decides whether to revise animation, strengthen controls, or regenerate a masked interval. Preserve accepted regions where the editor permits it and recheck boundary frames after patching. Mark an event unknown when the final view lacks sufficient evidence, and include unknowns in the reported failure count.

Validation plan

Experimental design

Create 150 prompts spanning rigid tabletop demonstrations with two to five events, including minimal pairs that swap order, change an object binding, or add an exclusion. Hold out combinations of relation types and objects. Compare direct prompting, one-shot Blender conditioning, multi-round free-form critique, and the constraint-driven loop using the same video editor and generation-call ceiling. Ablate the symbolic check and the pixel check separately. Include a Blender-only reference to measure control lost at neural rendering.

Outcomes and rejection criteria

Primary outcome is all-constraint satisfaction on final clips, assessed by blinded raters who see the prompt and output but no internal trace. Compare automated event labels against human annotations and report verification precision, recall and abstention. Measure visual quality, boundary artifacts, wall time and retries. Cluster intervals by prompt family. The core hypothesis fails if improvements appear in previews but disappear in final pixels, or if comparable quality and correctness are achieved by budget-matched free-form critique.

Feasibility and deliverables

Weeks 1-2 implement the event vocabulary and Blender preview pipeline. Weeks 3-4 integrate a frozen control-capable editor and visual tools. Weeks 5-7 run paired evaluations and release scripts, predicate manifests and examples. Scope the prototype to visually measurable rigid-object events in 4-8 second clips. If hard events remain invisible, report the failure by evidence availability and evaluate a separate camera-placement intervention.

References

[1] VideoDirectorGPT: Consistent Multi-scene Video Generation via LLM-Guided Planning. arXiv:2309.15091.

[2] VideoCoCo: Code-as-CoT for Physically-Consistent Video Generation via an Agentic Dual-Engine System. arXiv:2607.27380.

[3] VACE: All-in-One Video Creation and Editing. arXiv:2503.07598.