SEAM: Shot Entity-Attribute Memory for Consistent Short-Drama Generation at Scale

Jiaqi Liu
Shanghai Jiao Tong University
Shanghai, China
jkliu189@gmail.com

Maolin Ran
Shanghai Jiao Tong University
Shanghai, China
maolinr03@sjtu.edu.cn

Xiaoyang Lu
Shanghai Jiao Tong University
Shanghai, China
xiaoyangl@sjtu.edu.cn

Jian Wang
CreativeFitting
Shanghai, China
jim.wang@creativefitting.ai

Weiwen Liu*
Shanghai Jiao Tong University
Shanghai, China
wwliu@sjtu.edu.cn

Jianghao Lin
Shanghai Jiao Tong University
Shanghai, China
linjianghao@sjtu.edu.cn

Yong Yu
Shanghai Jiao Tong University
Shanghai, China
yyu@sjtu.edu.cn

Weinan Zhang*
Shanghai Jiao Tong University
Shanghai, China
wnzhang@sjtu.edu.cn

Abstract

Short-drama generation has grown into a large, industrialized pipeline, and as it scales from isolated shots to the episode level, visual continuity has become a critical bottleneck. Current agent frameworks generate each shot in isolation, so context drifts across shots, and props, character posture, and blocking turn inconsistent. Once assembled, these small discrepancies amplify into severe visual breaks. We present SEAM (**S**hot **E**ntity-**A**ttribute **M**emory), a training-free, model-agnostic memory graph that repairs continuity entirely at the prompt-text layer by extracting a multi-dimensional state for every shot, retrieving only causally prior context over the resulting graph, filtering it selectively, and injecting the surviving constraints by natural-language prompt rewriting. We further release SEAM-Bench, a double-blind continuity storyboarding benchmark, on which SEAM raises cross-episode continuity recall from 0.700 to 0.946, generalizes across six mainstream text models, and yields consistent, though not yet significant, gains at the generated-image layer. Deployed as a mandatory stage in CreativeFitting's SEAM-Agent production pipeline over 201 shots, SEAM reaches a 96.5% director-acceptance rate with zero unsafe injections; a conservative counterfactual attributes at least 21.9 percentage points of that rate to its cross-episode memory.

CCS Concepts

• **Computing methodologies** → **Natural language generation**; *Image and video generation*; • **Information systems** → *Retrieval models and ranking*.

*Corresponding author.

KDD '27, San Jose, CA, USA

ACM ISBN 978-1-4503-XXXX-X/2027/08
https://doi.org/XXXXXXX.XXXXXXX

Keywords

Memory graph, Short-drama generation, Retrieval augmentation

ACM Reference Format:
Jiaqi Liu, Maolin Ran, Xiaoyang Lu, Jian Wang, Weiwen Liu, Jianghao Lin, Yong Yu, and Weinan Zhang. 2027. SEAM: Shot Entity-Attribute Memory for Consistent Short-Drama Generation at Scale. In *Proceedings of the 33rd ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD '27), August 2027, San Jose, CA, USA.* ACM, New York, NY, USA, 14 pages. https://doi.org/XXXXXXX.XXXXXXX

1 Introduction

Short drama, a form of vertical, fast, multi-episode video fiction, is one of the fastest-growing segments of digital entertainment, and its production is now being reshaped by generative AI [10, 39, 40]. Its industrial workflow is highly standardized, spanning script writing, storyboarding, keyframe generation, and video synthesis, and agent frameworks increasingly absorb the repetitive labor around a professional director rather than replace the creative core [34]. The commercial stakes are already concrete: fully AI-generated titles now compete with live-action ones for the same audience, as on CreativeFitting's Reel.AI, one such platform serving AI-generated short dramas worldwide and the production setting we study in this paper. What makes the domain distinctive is its scale: a single title routinely spans tens of episodes with tens of shots each, so one production comprises thousands of shots that must remain mutually coherent.

At this scale, *visual continuity* is the central bottleneck, because each shot's prompt is generated in isolation and carries no persistent state memory. As shown in Figure 1, per-shot keyframe generation loses the glove that Shot 0 removed and lets it reappear in Shots 4 and 7; injecting the prior state from SEAM's memory graph carries the bare-hand state forward, and both shots stay consistent. Generators follow their prompts faithfully, so they amplify rather than absorb such contradictions, and over thousands of shots the drift accumulates into visual breaks a human editor must repair by hand.

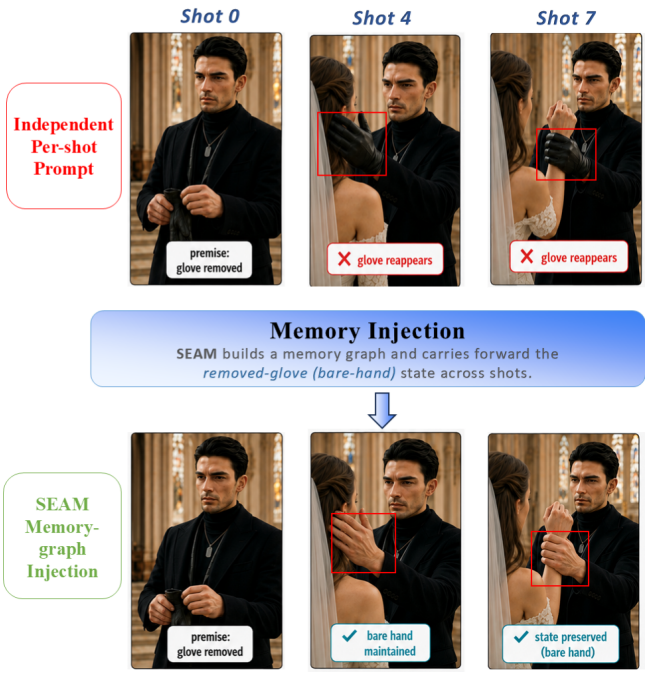

Figure 1: Cross-shot continuity break vs. SEAM repair. Top: per-shot independent generation loses an earlier state (a removed glove reappears), a contradiction no single shot can catch. Bottom: SEAM injects the prior state from its memory graph into the shot prompt, carrying it forward so the contradiction disappears.

Prior work enforces consistency inside a specific generator [2, 14, 15, 24], anchors it to 3D scenes or to video generated end to end [20, 43], or reuses generic retrieval and memory systems built for factual question answering [4, 12, 42]. All of them leave the storyboard prompt layer unaddressed, yet that layer is where the repair belongs: it sits upstream of every generator and stays readable and editable by the director. Repairing continuity there, inside a live production pipeline, raises three challenges that existing methods do not resolve. *C1: Heterogeneous, evolving visual state.* A shot's continuity depends on many dimensions at once (scene, atmosphere, characters and their states, spatial blocking, props, actions, camera style), each carrying forward or changing independently as the story progresses; unstructured text memory cannot represent this structured, multi-dimensional state. *C2: Causal and selective reuse.* Only *prior* state may be reused, so that causality is preserved, and only the truly relevant fragments should be injected, since indiscriminate copying overrides the director's creative intent. *C3: Generality and pluggability.* Generative backbones iterate rapidly, so a deployable remedy cannot be rebuilt whenever the underlying image or video model is replaced. It must be training-free, model-agnostic, pluggable as a standalone stage, and able to add continuity without altering the shot grammar or visual intent the director specified.

Guided by these challenges, we cast continuity as a *retrieval-and-injection* problem over cross-shot shared state. For *C1*, we parse each shot into a multi-dimensional state node and connect nodes by temporal-adjacency, character-co-occurrence, and scene-co-occurrence edges, turning the evolving state into an explicit, queryable memory graph. For *C2*, we retrieve only the relevant prior state and let an LLM selectively decide which candidates to inject; the visual description is then rewritten naturally, so continuity is added while the shot grammar and the director's intent stay intact. For *C3*, the whole procedure operates purely at the prompt-text layer and touches neither the model weights nor the downstream backbone, so it plugs in as a standalone stage and keeps working unchanged when the image or video model is swapped out. We call this framework SEAM, a **S**hot **E**ntity-**A**ttribute **M**emory graph, and embed it as one stage of SEAM-Agent, a multi-agent storyboarding pipeline that any script-to-screen system can adopt and that we run in commercial production.

Contributions. This paper makes the following contributions:

- *We introduce SEAM, the first shot entity-attribute memory graph for storyboard-layer continuity.* To our knowledge this is the first work to repair short-drama continuity at the storyboard prompt layer: each shot becomes a multi-dimensional visual state node, the temporal, character, and scene relations governing continuity become typed edges, and cross-shot consistency becomes a retrieval-and-injection problem over this graph.
- *We instantiate it as a training-free, prompt-text-layer procedure* of three stages (state extraction with graph construction, causal retrieval with selective filtering, natural-rewrite injection) that touches neither model weights nor the downstream backbone, so the stage stays model-agnostic and survives backbone upgrades.
- *We release SEAM-Bench, a double-blind continuity storyboarding benchmark*[1] that standardizes continuity evaluation for industrial short-drama generation at both the prompt and image layers.
- *We validate SEAM in live commercial production.* Deployed as a mandatory stage of the SEAM-Agent pipeline in CreativeFitting's production system, it reaches a 96.5% director-acceptance rate over 201 shots, cutting the manual continuity inspection a polished AI short drama otherwise demands from directors—evidence that the design holds outside the benchmark.

Section 2 surveys related work; Section 3 formalizes the problem and the continuity criterion; Section 4 presents SEAM, its memory graph, and its integration into SEAM-Agent; Section 5 reports SEAM-Bench, the three-layer evaluation, and the online deployment evidence.

2 Related Work

2.1 Agentic Short-Drama Production

The rise of short drama has motivated dedicated datasets, generation pipelines, and evaluation protocols. Large script corpora pair screenplays with shooting scripts [35], while agents take complementary creative roles: director–actor collaboration for controllable script writing [9], personalized frameworks carrying a single sentence to a produced drama [33], and geometry-guided control of cinematography [47]. The same role decomposition scales to feature-length work, where multi-agent movie pipelines distribute director, screenwriter, and storyboard-artist roles across LLM or VLM agents to plan and render multi-scene video [10, 39, 40], with

[1] https://huggingface.co/datasets/Jackyqq/SEAM-Bench

parallel efforts supplying synchronized sound [37]. How to score the result is itself unsettled: these systems predominantly report human Likert ratings or a single LLM judge [34, 39], leaving conclusions resting on one evaluator family, whereas story-visualization benchmarks supply image-side protocols for character identity and scene consistency [13, 36, 49] and storyboard-level benchmarks have begun to probe cinematographic quality [21, 46]. *Delta.* These pipelines automate authoring and rendering end to end but treat consistency as a byproduct of the generator or of scene-level planning, leaving no explicit representation of what carries over between shots, and no benchmark evaluates *cross-episode* continuity at the storyboard text layer alongside the keyframes rendered from it. We isolate and repair continuity at the textual storyboard that directors and downstream tools actually consume, via a shot-level memory graph deployed as a pipeline stage rather than a standalone generator, and release SEAM-Bench to check a continuity claim at the prompt layer and the image layer at once.

2.2 Memory for Cross-Shot Consistency

A large body of work seeks visual consistency across shots or frames. One line bakes it into the *generator*: cinematic models render coherent multi-shot sequences in a single pass [24], memory-flow methods propagate state across long-video generation [15], and diffusion extensions carry identity through transition tokens, caches, or entity-grounded scheduling [16, 18, 23]. A second anchors generation to an external structure such as a storyboard or a memory pack for 3D scenes [20, 43]. A third works in the image domain, preserving character identity across story panels by extracting a reusable character [3], controlling multi-character layout [7], interleaving text with images [41], grounding identity referentially [2], or repairing inconsistent panels after the fact [1]; that such anchoring is load-bearing is confirmed by character-stable pipelines reporting catastrophic drops once it is removed [14]. A separate line asks instead *how* an agent should store and retrieve what it has seen, arguing that episodic memory is the missing piece over long horizons [26, 45] and structuring retrieval through graph-based RAG [12], associative reuse [42], or persistent multimodal memory [4, 31], with continuity checked reference-free through entity graphs [8] or NLI contradiction detection [17]. *Delta.* The first three lines operate on pixels, latents, or a specific generator and are largely training-based or backbone-coupled, hard to insert into a model-heterogeneous pipeline; the memory machinery is instead built for factual recall. Visual continuity differs in what must be stored and in what may be reused: the unit of memory is a shot's multi-dimensional visual state rather than a proposition, the relations governing reuse are temporal adjacency and character/scene co-occurrence rather than semantic relatedness, and reuse must be strictly causal. SEAM adopts the retrieval-and-injection view but instantiates it over these shot-level states one level earlier than the generator, at the storyboard prompt-text layer, staying training-free and model-agnostic.

3 Preliminaries

This section fixes the notation for short dramas and shots, states the generation task, and defines the continuity criterion our core metric is built on. The memory graph is the heart of our design rather than background, so it is defined where it is used (Section 4.2.1).

Dramas and shots. A short drama is an episode-ordered set $\mathcal{D} = \{E_1, \ldots, E_M\}$ of M episodes, whose m-th episode $E_m = (s_1, \ldots, s_{n_m})$ is a temporally ordered sequence of n_m shots ($1 \le m \le M$). Each shot s_i is a structured tuple

$$s_i = (\ell_i,\ \kappa_i,\ x_i,\ t_i,\ C_i,\ P_i), \tag{1}$$

where $\ell_i \in \mathcal{L}$ is the scene/location, κ_i the shot grammar (shot size, angle, movement), x_i the visual description, t_i the dialogue, and $C_i \subseteq \mathcal{C}$, $P_i \subseteq \mathcal{P}$ the characters and props active in the shot. Here $\mathcal{C}$, $\mathcal{P}$ and $\mathcal{L}$ are the drama-wide universes of characters, props and scenes, respectively; the shot index i runs within an episode, and where cross-episode context matters it is read in the global shot order that $\mathcal{D}$ induces.

Generation task. The pipeline produces, for each shot, a prompt p_i that a downstream image/video generator consumes. Writing f_θ for the shotlist-authoring model with parameters θ (an LLM backbone in all our experiments), the per-shot, stateless baseline is

$$p_i = f_\theta(s_i), \tag{2}$$

that is, the prompt depends only on the current shot and carries no cross-shot state. This is the condition our memory stage is measured against.

Continuity and continuity recall. Continuity concerns the *persistent entities* of a drama—its characters, props and scenes—whose visual attributes should stay consistent across shots unless the narrative motivates a change. For an entity $e \in \mathcal{C} \cup \mathcal{P} \cup \mathcal{L}$ we write $\pi_i(e)$ for its *state projection*: the visual attributes (character appearance, prop possession, scene layout) that shot s_i ascribes to e, with $\pi_i(e) = \varnothing$ when e is inactive in that shot. For a shot pair $i < j$ in which e is active in both, a *continuity defect* $\delta(e, i, j) = 1$ is recorded when $\pi_i(e)$ contradicts $\pi_j(e)$ without narrative motivation. This grounds the *continuity recall*

$$\text{Recall} = \frac{\#\{\text{defects successfully repaired}\}}{\#\{\text{defects judged in need of repair}\}}, \tag{3}$$

our core metric for repairing cross-shot and cross-episode continuity (measured in Section 5).

4 Methodology

In this section, we introduce our **S**hot **E**ntity-**A**ttribute **M**emory graph for short-drama storyboarding (i.e., SEAM).

4.1 Overview

Guided by the three challenges of Section 1 (*C1* heterogeneous evolving state, *C2* causal selective reuse, *C3* training-free model-agnostic delivery), our methodology has two parts: *how continuity is represented and repaired*, and *how the repair is delivered inside a live production system.* For the first, we design SEAM, a cross-shot continuity memory graph that turns the retrieval-and-injection view of continuity into an executable, prompt-text-layer procedure: from the episode *script* we extract a structured multi-dimensional visual state for every shot and link the states into a directed memory graph whose temporal, character, and scene edges govern continuity; when the shotlist is (re-)authored, we retrieve only the causally

prior context, filter it selectively to discard redundant or conflicting fragments, and rewrite the shot's visual description so the surviving continuity is woven in while the shot grammar and the director's intent stay intact. SEAM runs in an *online* LLM-driven mode, used for every result reported in this paper, and a deterministic *offline* mode that substitutes rules for each LLM call so the stage still runs where no model is reachable. For the second, we embed SEAM into SEAM-Agent, our multi-agent storyboarding pipeline, as a mandatory "memory-optimizer" stage that builds the graph from the script, consumes the upstream shotlist, and repairs continuity shot by shot before rendering; because it reads the script and rewrites only the textual shotlist, it depends on neither the upstream authoring model nor the downstream backbone.

4.2 SEAM: A Cross-Shot Continuity Memory Graph

As shown in Figure 2, SEAM is organized as three stages operating entirely at the prompt-text layer: state extraction and graph construction, graph retrieval with selective filtering, and natural-rewrite injection. We describe each in turn.

4.2.1 State Extraction and Graph Construction. Whereas the state projection π_i of Section 3 tracks one entity at a time, our memory represents a whole shot at once. The extractor takes the episode script as input and, for every shot s_i it induces (Eq. (1)), produces a d-dimensional visual state $\boldsymbol{\sigma}_i = (\sigma_i^1, \ldots, \sigma_i^d)$, whose $d = 8$ dimensions cover scene, atmosphere, characters, character states, spatial blocking, props, actions, and camera style (Table 9). We formulate the extraction as an operator

$$\boldsymbol{\sigma}_i = \Phi(s_i), \tag{4}$$

which is instantiated in the online mode by prompting an LLM with a fixed dimension template that returns one structured record per shot, and, in the offline mode, by deterministic rule-based parsing of the same script. Heterogeneous sources are first normalized into a unified shot stream, so Φ applies uniformly regardless of the source format, and the dimension set is configurable rather than fixed by the formulation; $d = 8$ is the instantiation used throughout this paper.

Over the resulting states we construct a directed *memory graph* $G = (V, E)$. Its nodes V are the shot-state nodes v_i (each carrying $\boldsymbol{\sigma}_i$) together with resource-entity nodes materialized for the recurring characters, props, and scenes, and we connect nodes through the three relation types that carry continuity: a temporal-adjacency relation linking a shot to its neighbors within a window $|i - j| \le N$, a character-co-occurrence relation linking shots that share a character ($C_i \cap C_j \neq \varnothing$), and a scene-co-occurrence relation linking shots set in the same place ($\ell_i = \ell_j$). Each edge (i, j) carries a continuity weight $w_{ij} \in [0, 1]$ that grades how strongly the earlier shot constrains the later one, from a hard must-continue link ($w_{ij} = 1$) down to a weak association. In the online mode the relation type and weight of every edge are inferred by an LLM over overlapping shot windows and then deduplicated by keeping, for each ordered pair and type, the edge of maximum weight; in the offline mode the same three relations are recovered by deterministic co-occurrence and adjacency rules. These relations jointly determine which earlier shots are eligible to constrain the current one (Figure 3(a)), and the weights w_{ij} drive the weighted, multi-hop expansion used during retrieval.

4.2.2 Graph Retrieval and Selective Filtering. To repair shot s_i, we retrieve a continuity context $\mathcal{K}_i = \mathrm{Retrieve}(G_{<i}, s_i)$ from the *prior* subgraph $G_{<i}$, the part of G induced by nodes with index $j < i$. This enforces causality: a shot may inherit state from what has already happened, never from the future, so no forward leakage can occur.

We support two complementary retrieval modes. When a shot carries explicit resource references, we take its prior state by *resource overlap*: writing $R_i = C_i \cup P_i \cup \{\ell_i\}$ for its resource set, we collect the prior nodes $\{v_j \in G_{<i} \mid R_j \cap R_i \neq \varnothing\}$ and keep the most recent states of the shared characters, props, and scenes; when no prior node overlaps, we fall back to the last few preceding nodes so the context is never empty. For arbitrary inputs without such references, we instead perform *text-matched retrieval with edge expansion*, first scoring every prior node $v_j \in G_{<i}$ by a lightweight lexical overlap with s_i,

$$\mathrm{sim}(s_i, v_j) = \min\Big(1,\ \frac{|T(s_i) \cap T(v_j)|}{|T(s_i)|} + \beta[\, C_i \cap C_j \neq \varnothing\,]\Big), \tag{5}$$

where $T(\cdot)$ is a token set that mixes whole words with character-level bigrams so that the score stays robust across languages, the first term is the recall of the query tokens, and the second term adds a bonus β when the two shots share a named character. We keep the top-k prior nodes with sim > 0 as seeds $\mathcal{S}_i$, and then expand h hops *backward* along the temporal, character, and scene relations, traversing only edges whose weight clears a threshold $w_{ij} \ge \tau$,

$$\mathcal{K}_i = \bigcup_{v \in \mathcal{S}_i} \mathrm{Expand}^{\mathrm{bwd}}_{\le h}(v;\, G_{<i}, \tau), \tag{6}$$

where Expand collects the nodes reachable within h backward hops. The backward-only direction over $G_{<i}$ enforces causality at the graph level, while τ prunes weak associations so expansion does not drown the relevant context. The expansion recovers entities absent from the immediate neighbors yet continuous over a longer range (Figure 3(b)); with no explicit edges it falls back to implicit links induced on the fly. Our lexical scoring follows the probabilistic relevance tradition of BM25 [29] but uses no embeddings, so retrieval adds no model dependency.

Not every retrieved candidate should be injected. Indiscriminate concatenation of $\mathcal{K}_i$ would overwrite legitimate creative change and pollute the prompt with redundant or conflicting context. We therefore apply a *selective filtering* step that judges each candidate independently. For a candidate $c \in \mathcal{K}_i$, a decision function returns a keep/discard indicator together with a justification,

$$\phi(c, s_i) \in \{0, 1\}, \qquad \mathcal{K}_i^* = \{\, c \in \mathcal{K}_i \mid \phi(c, s_i) = 1 \,\}, \tag{7}$$

where ϕ is realized in the online mode by an LLM that scores candidates against s_i under an explicit rubric, discarding one that stands on its own, is redundant, or is separated from s_i by a scene cut, and keeping one only when it supplies genuine cross-shot continuity, with an auditable justification. In the offline mode ϕ degrades to a conservative rule that keeps a candidate whenever the shot has any prior context. Retaining only $\mathcal{K}_i^*$ is what makes the memory *selective*.

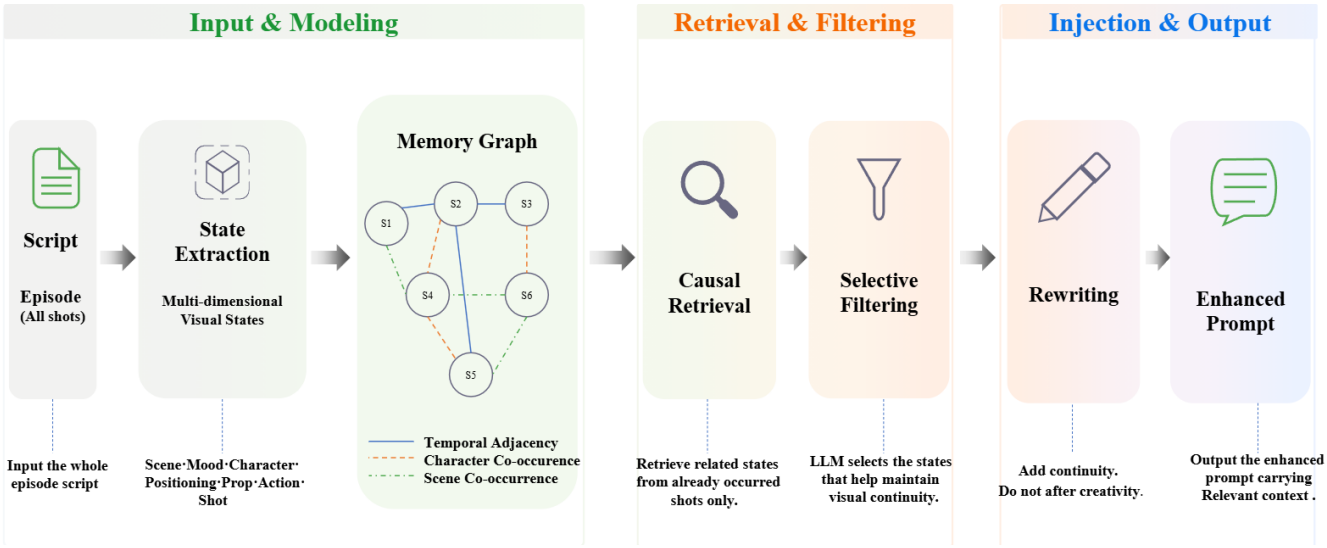

Figure 2: SEAM overview. The episode script is parsed into multi-dimensional shot states, which are linked into a memory graph by temporal, character, and scene edges. For each shot of the upstream shotlist, causally prior context is retrieved, filtered selectively, and rewritten into the visual description, yielding a continuity-enriched prompt.

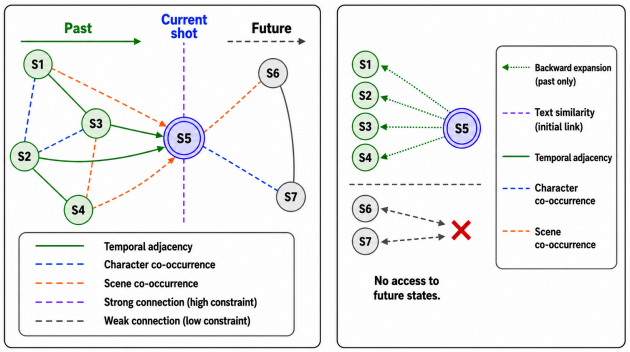

Figure 3: (a) The three relation types (temporal, character, and scene). (b) Graph retrieval with backward edge expansion over the prior subgraph $G_{<i}$.

4.2.3 Natural-Rewrite Injection. Given the filtered context $\mathcal{K}_i^*$, an injection operator g produces the continuity-enriched prompt

$$p_i = g(s_i, \mathcal{K}_i^*), \tag{8}$$

which replaces the stateless $f_\theta(s_i)$ of Eq. (2). We realize g in two modes. The preferred mode is a *natural rewrite*: an LLM weaves the retained continuity into the visual description x_i in the director's own register, adding only the continuity constraints while leaving the shot grammar κ_i and the visual semantics unchanged. Because an unconstrained rewrite could silently drift from the director's intent, we accept the rewritten description x_i' only when it passes a guard $\mathcal{V}(x_i', x_i)$ that (i) bounds the length ratio $|x_i'|/|x_i| \in [\rho_{\min}, \rho_{\max}]$, (ii) preserves every resource reference of x_i, and (iii) keeps its structural markers; a rewrite failing the guard is rejected and the shot reverts to its previous description. As a lightweight fallback, a *mechanical injection* mode appends $\mathcal{K}_i^*$ at a fixed prompt position without an LLM. Both modes add continuity rather than rewriting intent, and the guard preserves the director's original creative decisions.

4.3 SEAM-Agent: SEAM Within a Multi-Agent Pipeline

In this section, we describe how SEAM is deployed as one stage of SEAM-Agent. As shown in Figure 4, SEAM-Agent is a serial multi-agent pipeline in which the agents communicate through shotlist artifacts on disk, and SEAM operates as a memory optimizer between authoring and rendering.

4.3.1 Pipeline Structure. SEAM-Agent comprises four serial stages. A **Director Agent** reads the episode script and drafts the overall scene intent and narrative beats; a **Cinematographer Agent** turns that intent into a concrete shotlist, assigning the shot grammar κ_i (shot size, angle, and movement) to each shot; the **SEAM Memory Optimizer** performs cross-shot continuity repair over that shotlist; and a **Formatter** standardizes the result into the tabular schema that downstream tools consume. Each stage writes an artifact that the next stage reads, so the chain is decoupled at the file level. SEAM is a mandatory stage in this chain, not an optional pass that runs afterward, so every shotlist the pipeline emits has been continuity-checked before the downstream keyframe and video generators consume it.

4.3.2 SEAM as the Memory-Optimizer Stage. The memory optimizer takes two inputs: the episode script, from which it builds and persists the memory graph, and the upstream shotlist, whose descriptions it repairs. Applying the three-stage procedure of Section 4.2, it extracts the shot states into G, retrieves and filters each shot's prior context over $G_{<i}$, and rewrites the description via Eq. (8), emitting $p_i = g(s_i, \mathcal{K}_i^*)$ in place of the stateless baseline $f_\theta(s_i)$ of Eq. (2). Because state accumulates across episodes, the retrieved context spans earlier shots of the current episode and

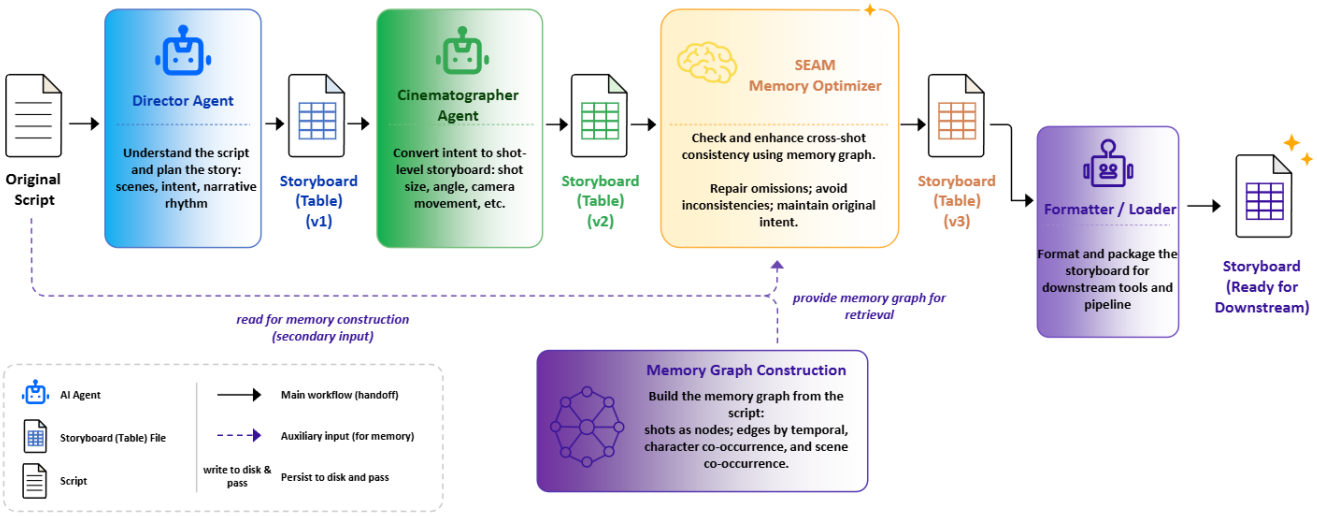

Figure 4: SEAM sits as the "memory optimizer" stage, placed after the authoring agents and before the loader. It builds the memory graph from the episode script and repairs continuity shot by shot over the shotlist that the authoring agents have already generated.

prior episodes alike, which is what lets the optimizer repair the cross-episode breaks that dominate long titles. Since it operates solely at the prompt-text layer, it is decoupled from both the authoring model and the downstream backbone, a property we verify with the six-model results of Section 5.

5 Experiments

We study three research questions. **Q1**: does the memory graph repair the cross-shot and cross-episode continuity defects a stateless pipeline leaves behind? **Q2**: is this repair model-agnostic across heterogeneous backbones? **Q3**: do prompt-layer repairs survive the text-to-image stage and stay measurable in the keyframes?

5.1 Experimental Setup

SEAM-Bench. As summarized in Table 1, SEAM-Bench is a short-drama continuity storyboarding benchmark over three produced dramas totaling 68 episodes, referred to throughout by the short identifiers of the released data: `his-toyboy` (*His Toyboy: The Billionaire's Trap*), `beyond-the-wall` (*Beyond the Wall*), and `werewolf` (*You Are My Cure, My Undoing*). We release two kinds of material: the original scripts forming the pipeline input s_i, and reference images—character/scene visual anchors from a semantically renamed, deduplicated pool—serving as fixed keyframe input and as the image-layer gold standard. Expert human-director storyboards annotated shot by shot are the professional reference, but remain restricted by copyright, held out from the release, and used only for evaluation and unblinding.

Setup. We evaluate six heterogeneous text models (`claude`, `deepseek-v4-pro`, `glm-5.1`, `gpt-5.4`, `kimi-k2.6`, `minimax-m2.7`). Per episode, each backbone produces an uninjected *shotlist1* (Eq. (2)) and a memory-injected *shotlist2* (Eq. (8)). The two CSVs are column- and shot-aligned, making every comparison a paired within-model contrast whose only varying factor is memory injection. Keyframes

Table 1: SEAM-Bench data statistics. AI shots are counted on the `claude` run; other backbones differ by $< 5\%$.

Drama	Ep.	Human	AI	Ref. img.
`his-toyboy`	23	754	907	15
`beyond-the-wall`	20	659	814	31
`werewolf`	25	826	997	12
Total	68	2,239	2,718	58

come from Nano Banana Pro on both variants under identical references and template skeletons, from which we extract 900 test instances for the image layer, of which 820 returned an image. Scoring is delegated to Gemini 3 Pro, excluded from the six evaluated backbones so that no model judges its own storyboards.

Evaluation protocol. Storyboarding has no established automatic text-layer metric, and recent script-to-screen systems rely on human Likert ratings or LLM judges alone [34, 39]. To keep no conclusion resting on a single evaluator family, we pair a double-blind LLM judge with a CPU-only offline metric suite assembled from adjacent literatures [5, 8, 17, 19, 21, 22, 28, 44, 46]. The suite scores a storyboard along four complementary axes: agreement with the human director, accuracy of the cinematography fields, camera-label distribution, and reference-free continuity. An image layer then scores the rendered keyframes against their references [6, 13, 25, 27, 30, 32, 36, 48, 49]. Definitions and formulas are deferred to Appendix A, where Table 8 lists every abbreviation used below. In the double-blind judge, conditions appear as "A/B" with sources hidden and deterministically counterbalanced, revealed only after scoring; the prompt layer is scored on five dimensions and the image layer on four. Every baseline-vs-memory contrast is paired at the episode level and tested with the Wilcoxon signed-rank test under Holm correction [11, 38] (*: corrected $p < 0.05$). The 80 renders refused by the provider's safety filter (triggered by sensitive plot content, near-symmetric across conditions: 412 baseline vs.

Table 2: Q1 continuity recall (his-toyboy, 23 episodes). The control restricts retrieval to the current episode, so it surfaces fewer judgeable defects; #Judged is therefore part of the effect rather than a fixed denominator (see text).

Condition	Recall↑	#Judged	#Modified
Control (episode-local retrieval)	0.700	10	7
SEAM (cross-episode graph)	0.946	37	35

408 memory images survive) are dropped pairwise, so a shot enters the image metrics only if both conditions rendered it, leaving 403 paired shots. The NLI and image layers run on beyond-the-wall, the rest on all three dramas.

5.2 Main Results (Q1/Q2/Q3)

Q1: memory injection repairs continuity. Three measurements triangulate the repair effect. First, we turn to *continuity recall* (Eq. (3)). As shown in Table 2, on his-toyboy (23 episodes) the memory graph repairs 35 of the 37 defects the judge rules in need of repair (0.946), against 7 of 10 (0.700) for a control whose retrieval is restricted to the current episode. The two conditions do not share a denominator, and the reason is itself part of the result: a defect can only be judged once retrieval surfaces the prior state it contradicts, so the episode-local control yields 3.7× fewer candidate sites. SEAM therefore improves on two axes at once, exposing 27 additional genuine defects and repairing a larger fraction of those it exposes; the recall column understates the gap, since those 27 are left unrepaired rather than counted against the control. Resting on different defect populations, this measurement establishes that cross-episode retrieval is what makes defects addressable, and we rely on the next two, both computed per shot, for the magnitude of the repair. Second, the *targeted state-conditioned contradiction* probe scores each filter-flagged shot (Section 4.2.2) against its graph-retrieved expected state as NLI premise, before versus after injection. Table 3 reports the outcome: across all 2,946 flagged shots, mean contradiction probability drops from 0.308 to 0.131 and the threshold-exceeding fraction from 0.264 to 0.093, removing roughly two thirds of the contradiction mass at the targeted sites in every drama–backbone cell, with a residual 4–7% resisting repair that Figure 5 exposes and Section 5.4 takes up. Third, the *double-blind prompt-layer judgment* over six backbones (30 model-episodes, episodes 1–5 of beyond-the-wall, the same subset carrying the image layer) favors memory on all five dimensions. As Table 5 and Figure 8 show, character-state continuity (+4.4, r=0.82), prop continuity (+2.9, r=0.77), and intent fit (+6.5, r=0.90) stay significant after Holm correction.

Q2: model-agnosticism. Table 4 consolidates the per-model evidence across the three metric families. Recall (panel a) is high in all 18 drama–backbone cells, from 0.667 (claude on his-toyboy) to 1.000 (minimax on beyond-the-wall), with per-model means spanning 0.731–0.992. The targeted contradiction drop (panel b) holds throughout, mean probability falling from the 0.29–0.35 band to 0.10–0.17. The blind gains (panel c) are positive on 17 of 18 model–dimension entries; the sole exception, prop continuity for glm-5.1, is a −0.04 shift rounding to zero. The spread reflects each model's style, yet the direction never reverses, evidence of model-agnosticism.

Table 3: Targeted state-conditioned contradiction on repair-flagged shots. The premise is the graph-expected entity state and the score is the NLI contradiction probability before vs. after injection, pooled over six backbones per drama and weighted by flagged-shot count.

		Mean prob.↓		Rate ($p \geq 0.5$)↓	
Drama	#Flag.	before	after	before	after
beyond-the-wall	1,085	0.344	0.106	0.320	0.069
his-toyboy	923	0.302	0.157	0.255	0.122
werewolf	938	0.273	0.133	0.209	0.091
All	2,946	0.308	0.131	0.264	0.093

Table 4: Per-model results over the six backbones in three metric groups: *(a)* continuity recall per drama—BTW (beyond-the-wall), HTY (his-toyboy), WLF (werewolf)—and its mean; *(b)* targeted contradiction probability before/after injection, pooled over dramas and weighted by flagged-shot count (n); *(c)* blind prompt-layer gains Δ=M−B on the memory-edited dimensions (Table 8). Every cell improves, answering Q2 in the affirmative.

	(a) Recall↑				*(b) Contradiction*			*(c)* Δ*Blind*↑		
Backbone	BTW	HTY	WLF	**mean**	n	bef.↓	aft.↓	CSC	PRP	IFIT
claude	0.760	0.667	0.767	**0.731**	178	0.321	0.129	+5.1	+0.9	+4.9
deepseek	0.912	0.959	0.915	**0.929**	863	0.286	0.115	+9.9	+5.9	+15.8
glm-5.1	0.983	0.947	0.960	**0.963**	248	0.308	0.116	+3.5	−0.0	+2.8
gpt-5.4	0.994	0.988	0.995	**0.992**	548	0.294	0.135	+2.1	+1.1	+2.8
kimi	0.961	0.928	0.960	**0.950**	399	0.353	0.103	+2.7	+1.0	+7.6
minimax	1.000	0.967	0.993	**0.987**	710	0.319	0.167	+3.0	+8.8	+4.8

The ranking is not an artifact of defect volume. Judged defects per episode vary by an order of magnitude across cells (1.7–19.8, or 3.6–13.7 per model), yet five of six per-model means land in a tight 0.93–0.99 band. claude is the sole outlier, lowest on recall not because it judges more defects (it flags the *fewest*) but because its rewrites are the most conservative, and recall counts a defect repaired only when a changed description is emitted; on the blind judgment it still posts among the largest character-state gains (+5.1). Since recall cannot separate "no rewrite" from "wrong rewrite," we triangulate with the targeted probe and the blind judge.

Q3: transfer to the image layer. As Table 5b shows, at the generated-image layer the blind judge still prefers memory on the two dimensions injection actually edits, character appearance (+3.7) and prop continuity (+3.3), while scene layout and frame quality stay flat. With only n=30 sequence pairs these deltas do not reach significance, so we answer Q3 only directionally: transfer is visible but attenuated by the stochastic text-to-image stage.

Offline metrics: what moves and what does not. Table 6 shows how the offline suite separates *targeted repair* from *global drift*. Against the human director the paired shifts sit in the third decimal and the divergences stay at zero, since injection edits entity-state phrases without touching the cinematography fields, while the reference-free layer moves in the expected direction on every axis; we return to this contrast in Section 5.4, where the two-thirds targeted drop quantifies it. The offline image embeddings show the same attenuation, all six scores flat at this sample size, consistent with generation noise dominating.

Table 5: Double-blind LLM judgment (0–100) per dimension, pooled over six backbones (`beyond-the-wall`, ep. 1–5): *B* baseline, *M* memory, Δ=M−B (* Holm-corrected $p < 0.05$); n=30 model-episodes and 30 sequences.

(a) Prompt layer (per-shot)

	CSC↑	SCON↑	PRP↑	CGEN↑	IFIT↑
B	88.1	91.8	92.7	91.1	87.8
M	92.5	92.9	95.7	91.7	94.2
Δ	+4.4*	+1.1	+2.9*	+0.6	+6.5*

(b) Image layer (per-sequence)

	CAPP↑	SLAY↑	PRPI↑	SFQ↑
B	64.0	84.3	62.3	93.0
M	67.7	83.1	65.7	94.0
Δ	+3.7	−1.2	+3.3	+1.0

Table 6: Offline text-layer metrics over 408 model-episodes, reference-based (top) and reference-free (bottom): *s1* baseline, *s2* memory, Δ=s2−s1 (* Holm-corrected $p < 0.05$); ↑/↓ is the improvement direction. JS pools the three divergences, bit-identical across conditions; NLI runs on `beyond-the-wall` (n=192).

	SM-F1↑	BS-F1↑	SZ-ACC↑	Kτ↑	JS↓
s1	0.4755	0.6379	0.318	0.446	fixed
s2	0.4731	0.6367	0.313	0.412	fixed
Δ	−0.002*	−0.001*	−0.005	−0.034*	0.000
	EOD↑	ACOS-M↑	ACOS-N↑	NLI-CR↓	NLI-MP↓
s1	11.06	0.4860	0.148	0.668	0.656
s2	11.06	0.4908	0.148	0.660	0.649
Δ	0.000	+0.005*	0.000	−0.009*	−0.007*

5.3 Online Deployment Evidence for Memory Effectiveness

Beyond the offline suite, we report a live online run of SEAM as the mandatory memory-optimization stage of SEAM-Agent in CreativeFitting's production system, over a freshly produced short-drama of four consecutive episodes (201 shots), whose deployment architecture is detailed in Appendix B. Human annotators decide acceptance under a human-specified four-state standard, a proxy for *director acceptance*.

As summarized in Table 7, SEAM reached an overall director-acceptance rate of 96.5% at a zero unsafe-injection error rate, counting an injection unsafe if it contradicts the shot it edits or overwrites the director's stated intent. The residual misses share one failure mode: a persistent state is tracked correctly in some shots of a run but not all, so the injection is incomplete rather than wrong. To bound how much of the rate the memory explains, we compute a pessimistic counterfactual: reclassifying every accepted shot whose repair drew on a prior episode as a miss drops acceptance to 74.6%, a lower bound of $\Delta = 21.9$ pp on the contribution of cross-episode memory. It is conservative: it discards such a shot outright instead of crediting the episode-local continuity it would still have received. The mechanism is restrained: it injects into only 27.4% of shots, with minimal fragments (median ratio 27.6%) and non-injected shots unchanged to the byte, so memory adds continuity without rewriting intent, corroborating Q1 in production. These storyboards carry commercially released work: titles from this pipeline reach viewers on Reel.AI, where *Lost Before I Found You* has topped the DataEye overseas micro-drama heat ranking by a wide margin over the live-action titles that otherwise dominate it. Chart position is no controlled measurement and we claim no causal credit, but it does place the 96.5% on production work, not on a pilot.

Table 7: Online memory-effectiveness over the four deployed episodes. The counterfactual row is a *conservative lower bound*: every cross-episode-dependent repair reclassified as a miss.

Metric	Value
Shots evaluated	201
Director-acceptance rate↑	96.5%
w/o cross-episode memory (lower bound)↑	74.6%
memory attribution (Δ)↑	+21.9 pp
Selective memory-injection rate	27.4%
Injection edit locality (median)	27.6%
Unsafe-injection error rate↓	0.0%

5.4 Further Analysis

Targeted repair versus global dilution. As shown in Figure 7, which puts the two measurement scopes on a common relative scale, the contradiction rate and probability improve by 64.9% and 57.7% at the flagged sites, whereas every episode-level metric moves by at most 1.3%. The ratio indicates a sparse effect, not a weak one: the filter flags roughly one shot in ten, so an episode-level average spreads a near-total repair over an order of magnitude more unaffected shots, which is why we measure continuity at flagged sites and why Table 6 stays flat. Nor is the drop carried by a favorable subset: Figure 6 resolves Table 3 into all 18 drama–backbone cells, and the mean probability falls in every one, from 0.25–0.39 to 0.05–0.20, so the repair floor is set by the memory graph, not the authoring model.

Failure modes and limitations. We report the failures directly. *(i)* Not every judged defect is repaired: 37 judged versus 35 modified for Q1, and 4–7% of flagged shots stay above threshold, which Figure 5 makes visible as the mass above the diagonal. *(ii)* Image-layer transfer, as Table 5b reports, is directional but not significant at n=30, so Q3 rests on the weakest evidence of the three. *(iii)* Kendall's τ drops under injection (0.446 to 0.412, $p < 0.05$), a reordering side effect the rewrite guard does not constrain. *(iv)* Camera-angle accuracy stays near 0.20 in both conditions, so the shot-grammar gap remains open. Two limits further bound the evidence: one judge family supplies both the recall labels and the blind scores, mitigated but not removed by the offline suite; and the fallback modes of Section 4.2 are unevaluated.

6 Conclusion

We presented SEAM, a training-free, model-agnostic memory graph that repairs cross-shot and cross-episode continuity in prompt text. It lifts continuity recall from 0.700 to 0.946, stays positive across six text models (0.731–0.992), and transfers directionally to images. SEAM-Bench is released, and SEAM runs as a mandatory SEAM-Agent stage in production. Future work targets video-layer evaluation.

References

[1] Kiymet Akdemir, Tahira Kazimi, and Pinar Yanardag. 2025. Audit & Repair: An Agentic Framework for Consistent Story Visualization in Text-to-Image Diffusion Models. arXiv:2506.18900v1

[2] Aditya Arora, Akshita Gupta, Pau Rodriguez, and Marcus Rohrbach. 2026. ReCap: Lightweight Referential Grounding for Coherent Story Visualization. arXiv:2604.18575v1

[3] Omri Avrahami, Amir Hertz, Yael Vinker, Moab Arar, Shlomi Fruchter, Ohad Fried, Daniel Cohen-Or, and Dani Lischinski. 2024. The Chosen One: Consistent Characters in Text-to-Image Diffusion Models. In *ACM SIGGRAPH 2024 Conference Papers (SIGGRAPH '24)*. Association for Computing Machinery, New York, NY, USA, Article 26, 12 pages. doi:10.1145/3641519.3657430

[4] Chunliang Chen, Ming Guan, Xiao Lin, Jiaxu Li, Luxi Lin, Qiyi Wang, Xiangyu Chen, Jixiang Luo, Changzhi Sun, Dell Zhang, and Xuelong Li. 2025. TeleMem: Building Long-Term and Multimodal Memory for Agentic AI. arXiv:2601.06037v4

[5] Robin Courant, Nicolas Dufour, Xi Wang, Marc Christie, and Vicky Kalogeiton. 2025. E.T. the Exceptional Trajectories: Text-to-Camera-Trajectory Generation with Character Awareness. In *Computer Vision – ECCV 2024 (Lecture Notes in Computer Science, Vol. 15062)*. Springer, Cham, Switzerland, 464–480. doi:10.1007/978-3-031-73235-5_26

[6] Stephanie Fu, Netanel Tamir, Shobhita Sundaram, Lucy Chai, Richard Zhang, Tali Dekel, and Phillip Isola. 2023. DreamSim: Learning New Dimensions of Human Visual Similarity using Synthetic Data. In *Advances in Neural Information Processing Systems (NeurIPS)*, Vol. 36. Curran Associates Inc., Red Hook, NY, USA. doi:10.52202/075280-2208

[7] Yuan Gong, Youxin Pang, Xiaodong Cun, Menghan Xia, Yingqing He, Haoxin Chen, Longyue Wang, Yong Zhang, Xintao Wang, Ying Shan, and Yujiu Yang. 2023. Interactive Story Visualization with Multiple Characters. In *SIGGRAPH Asia 2023 Conference Papers (SA '23)*. Association for Computing Machinery, New York, NY, USA, Article 101, 10 pages. doi:10.1145/3610548.3618184

[8] Camille Guinaudeau and Michael Strube. 2013. Graph-based Local Coherence Modeling. In *Proceedings of ACL*. Association for Computational Linguistics, Sofia, Bulgaria, 93–103. https://aclanthology.org/P13-1010/

[9] Senyu Han, Lu Chen, Li-Min Lin, Zhengshan Xu, and Kai Yu. 2024. IBSEN: Director-Actor Agent Collaboration for Controllable and Interactive Drama Script Generation. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Association for Computational Linguistics, Bangkok, Thailand, 1607–1619. doi:10.18653/v1/2024.acl-long.88

[10] Liu He, Yizhi Song, Hejun Huang, Pinxin Liu, Yunlong Tang, Daniel Aliaga, and Xin Zhou. 2024. Kubrick: Multimodal Agent Collaborations for Synthetic Video Generation. arXiv:2408.10453v2

[11] Sture Holm. 1979. A Simple Sequentially Rejective Multiple Test Procedure. *Scandinavian Journal of Statistics* 6, 2 (1979), 65–70. https://www.jstor.org/stable/4615733

[12] Yiqian Huang, Shiqi Zhang, and Xiaokui Xiao. 2025. KET-RAG: A Cost-Efficient Multi-Granular Indexing Framework for Graph-RAG. arXiv:2502.09304v2

[13] 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. 2024. VBench: Comprehensive Benchmark Suite for Video Generative Models. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, Seattle, WA, USA, 21807–21818. https://openaccess.thecvf.com/content/CVPR2024/html/Huang_VBench_Comprehensive_Benchmark_Suite_for_Video_Generative_Models_CVPR_2024_paper.html

[14] Chayan Jain, Rishant Sharma, Archit Garg, Ishan Bhanuka, Pratik Narang, and Dhruv Kumar. 2025. Lights, Camera, Consistency: A Multistage Pipeline for Character-Stable AI Video Stories. arXiv:2512.16954v1

[15] Sihui Ji, Xi Chen, Shuai Yang, Xin Tao, Pengfei Wan, and Hengshuang Zhao. 2025. MemFlow: Flowing Adaptive Memory for Consistent and Efficient Long Video Narratives. arXiv:2512.14699v1

[16] Ozgur Kara, Krishna Kumar Singh, Feng Liu, Duygu Ceylan, James M. Rehg, and Tobias Hinz. 2025. ShotAdapter: Text-to-Multi-Shot Video Generation with Diffusion Models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, Nashville, TN, USA, 28405–28415. https://openaccess.thecvf.com/content/CVPR2025/html/Kara_ShotAdapter_Text-to-Multi-Shot_Video_Generation_with_Diffusion_Models_CVPR_2025_paper.html

[17] Philippe Laban, Tobias Schnabel, Paul N. Bennett, and Marti A. Hearst. 2022. SummaC: Re-Visiting NLI-based Models for Inconsistency Detection in Summarization. *Transactions of the Association for Computational Linguistics* 10 (2022), 163–177. doi:10.1162/tacl_a_00453

[18] Yixuan Lai, Tianjia Shao, Weijia Dou, Siyu Zhu, and Jingdong Wang. 2026. GroundShot: Visually Consistent Multi-Shot Long Video Generation via Entity-Grounded Shot Scheduling. arXiv:2606.20799v3

[19] Moritz Laurer, Wouter van Atteveldt, Andreu Casas, and Kasper Welbers. 2024. Less Annotating, More Classifying: Addressing the Data Scarcity Issue of Supervised Machine Learning with Deep Transfer Learning and BERT-NLI. *Political Analysis* 32, 1 (2024), 84–100. doi:10.1017/pan.2023.20

[20] Bingliang Li, Zhenhong Sun, Jiaming Bian, Yuehao Wu, Yifu Wang, Hongdong Li, Yatao Bian, Huadong Mo, and Daoyi Dong. 2026. StoryBlender: Inter-Shot Consistent and Editable 3D Storyboard with Spatial-temporal Dynamics. arXiv:2604.03315v1

[21] Yuzhi Li, Haojun Xu, and Feng Tian. 2025. From Shots to Stories: LLM-Assisted Video Editing with Unified Language Representations. arXiv:2505.12237v1

[22] Xiaoqiang Luo. 2005. On Coreference Resolution Performance Metrics. In *Proceedings of HLT/EMNLP*. Association for Computational Linguistics, Vancouver, British Columbia, Canada, 25–32.

[23] Xiangyang Luo, Qingyu Li, Xiaokun Liu, Wenyu Qin, Miao Yang, Meng Wang, Pengfei Wan, Di Zhang, Kun Gai, and Shao-Lun Huang. 2025. FilmWeaver: Weaving Consistent Multi-Shot Videos with Cache-Guided Autoregressive Diffusion. arXiv:2512.11274v1

[24] Yihao Meng, Hao Ouyang, Yue Yu, Qiuyu Wang, Wen Wang, Ka Leong Cheng, Hanlin Wang, Yixuan Li, Cheng Chen, Yanhong Zeng, Yujun Shen, and Huamin Qu. 2025. HoloCine: Holistic Generation of Cinematic Multi-Shot Long Video Narratives. arXiv:2510.20822v1

[25] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, Mahmoud Assran, Nicolas Ballas, Wojciech Galuba, Russell Howes, Po-Yao Huang, Shang-Wen Li, Ishan Misra, Michael Rabbat, Vasu Sharma, Gabriel Synnaeve, Hu Xu, Hervé Jégou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. 2024. DINOv2: Learning Robust Visual Features without Supervision. *Transactions on Machine Learning Research* (2024). https://openreview.net/forum?id=a68SUt6zFt

[26] Mathis Pink, Qinyuan Wu, Vy Ai Vo, Javier Turek, Jianing Mu, Alexander Huth, and Mariya Toneva. 2025. Position: Episodic Memory is the Missing Piece for Long-Term LLM Agents. arXiv:2502.06975v1

[27] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. 2021. Learning Transferable Visual Models From Natural Language Supervision. In *Proceedings of the 38th International Conference on Machine Learning (Proceedings of Machine Learning Research, Vol. 139)*. PMLR, Virtual Event, 8748–8763. https://proceedings.mlr.press/v139/radford21a.html

[28] Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. In *Proceedings of EMNLP-IJCNLP*. Association for Computational Linguistics, Hong Kong, China, 3982–3992. doi:10.18653/v1/D19-1410

[29] Stephen Robertson and Hugo Zaragoza. 2009. The Probabilistic Relevance Framework: BM25 and Beyond. *Foundations and Trends in Information Retrieval* 3, 4 (2009), 333–389. doi:10.1561/1500000019

[30] Nataniel Ruiz, Yuanzhen Li, Varun Jampani, Yael Pritch, Michael Rubinstein, and Kfir Aberman. 2023. DreamBooth: Fine Tuning Text-to-Image Diffusion Models for Subject-Driven Generation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, Vancouver, BC, Canada, 22500–22510. https://openaccess.thecvf.com/content/CVPR2023/html/Ruiz_DreamBooth_Fine_Tuning_Text-to-Image_Diffusion_Models_for_Subject-Driven_Generation_CVPR_2023_paper.html

[31] Samarth Sarin, Lovepreet Singh, Bhaskarjit Sarmah, and Dhagash Mehta. 2025. Memoria: A Scalable Agentic Memory Framework for Personalized Conversational AI. arXiv:2512.12686v1

[32] Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, Patrick Schramowski, Srivatsa Kundurthy, Katherine Crowson, Ludwig Schmidt, Robert Kaczmarczyk, and Jenia Jitsev. 2022. LAION-5B: An open large-scale dataset for training next Generation Image-Text Models. In *Advances in Neural Information Processing Systems*, Vol. 35. Curran Associates, Inc., New Orleans, LA, USA. doi:10.52202/068431-1833

[33] Yufei Shi, Weilong Yan, Naixuan Huang, Yucheng Chen, Chenyu Zhang, Tao He, Si Yong Yeo, and Ming Li. 2026. One Sentence, One Drama: Personalized Short-Form Drama Generation via Multi-Agent Systems. arXiv:2605.22144v1

[34] Kunpeng Song, Tingbo Hou, Zecheng He, Haoyu Ma, Jialiang Wang, Animesh Sinha, Sam Tsai, Yaqiao Luo, Xiaoliang Dai, Li Chen, Xide Xia, Peizhao Zhang, Peter Vajda, Ahmed Elgammal, and Felix Juefei-Xu. 2025. Llama Learns to Direct: DirectorLLM for Human-Centric Video Generation. In *Proceedings of the 36th British Machine Vision Conference (BMVC)*. BMVA, Sheffield, UK. https://bmvc2025.bmva.org/proceedings/896/

[35] Jing Tang, Quanlu Jia, Yuqiang Xie, Zeyu Gong, Xiang Wen, Jiayi Zhang, Yalong Guo, Guibin Chen, and Jiangping Yang. 2024. SkyScript-100M: 1,000,000,000 Pairs of Scripts and Shooting Scripts for Short Drama. arXiv:2408.09333v2

[36] Yoad Tewel, Omri Kaduri, Rinon Gal, Yoni Kasten, Lior Wolf, Gal Chechik, and Yuval Atzmon. 2024. Training-Free Consistent Text-to-Image Generation. *ACM Transactions on Graphics* 43, 4, Article 52 (2024), 18 pages. doi:10.1145/3658157

[37] Zixuan Wang, Chi-Keung Tang, and Yu-Wing Tai. 2025. ReelWave: Multi-Agentic Movie Sound Generation through Multimodal LLM Conversation. arXiv:2503.07217v3

[38] Frank Wilcoxon. 1945. Individual Comparisons by Ranking Methods. *Biometrics Bulletin* 1, 6 (1945), 80–83. doi:10.2307/3001968
[39] Weijia Wu, Zeyu Zhu, and Mike Zheng Shou. 2025. Automated Movie Generation via Multi-Agent CoT Planning. arXiv:2503.07314v1
[40] Tianyidan Xie, Zhentao Huang, Mingjie Wang, Xin Huang, Jun Zhou, Minglun Gong, and Zili Yi. 2026. CineAGI: Character-Consistent Movie Creation through LLM-Orchestrated Multi-Modal Generation and Cross-Scene Integration. arXiv:2604.23579v1
[41] Shuai Yang, Yuying Ge, Yang Li, Yukang Chen, Yixiao Ge, Ying Shan, and Yingcong Chen. 2024. SEED-Story: Multimodal Long Story Generation with Large Language Model. arXiv:2407.08683v2
[42] Kai Zhang, Xinyuan Zhang, Ejaz Ahmed, Hongda Jiang, Caleb Kumar, Kai Sun, Zhaojiang Lin, Sanat Sharma, Shereen Oraby, Aaron Colak, Ahmed Aly, Anuj Kumar, Xiaozhong Liu, and Xin Luna Dong. 2025. AssoMem: Scalable Memory QA with Multi-Signal Associative Retrieval. arXiv:2510.10397v1
[43] Peixuan Zhang, Zijian Jia, Kaiqi Liu, Shuchen Weng, Si Li, and Boxin Shi. 2025. STAGE: Storyboard-Anchored Generation for Cinematic Multi-shot Narrative. arXiv:2512.12372v2
[44] Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2020. BERTScore: Evaluating Text Generation with BERT. In *International Conference on Learning Representations (ICLR)*. OpenReview.net, Addis Ababa, Ethiopia. https://openreview.net/forum?id=SkeHuCVFDr
[45] Zeyu Zhang, Xiaohe Bo, Chen Ma, Rui Li, Xu Chen, Quanyu Dai, Jieming Zhu, Zhenhua Dong, and Ji-Rong Wen. 2024. A Survey on the Memory Mechanism of Large Language Model based Agents. arXiv:2404.13501v1
[46] Mingzhe Zheng, Dingjie Song, Guanyu Zhou, Jun You, Jiahao Zhan, Xuran Ma, Xinyuan Song, Ser-Nam Lim, Qifeng Chen, and Harry Yang. 2025. CML-Bench: A Framework for Evaluating and Enhancing LLM-Powered Movie Scripts Generation. arXiv:2510.06231v1
[47] Hengji Zhou, Sijie Liu, Jianrun Chen, Xingchen Zou, Lianghao Xia, and Liqiang Nie. 2026. DramaDirector: Geometry-Guided Short Drama Generation. arXiv:2606.24107v2
[48] Jinsong Zhou, Yihua Du, Xinli Xu, Luozhou Wang, Zijie Zhuang, Yehang Zhang, Shuaibo Li, Xiaojun Hu, Bolan Su, and Ying-cong Chen. 2026. VideoMemory: Toward Consistent Video Generation via Memory Integration. arXiv:2601.03655v1
[49] Cailin Zhuang, Ailin Huang, Yaoqi Hu, Jingwei Wu, Wei Cheng, Jiaqi Liao, Hongyuan Wang, Xinyao Liao, Weiwei Cai, Hengyuan Xu, Xuanyang Zhang, Xianfang Zeng, Zhewei Huang, Gang Yu, and Chi Zhang. 2026. ViStoryBench: Comprehensive Benchmark Suite for Story Visualization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, Denver, CO, USA, 9455–9466. https://openaccess.thecvf.com/content/CVPR2026/papers/Zhuang_ViStoryBench_Comprehensive_Benchmark_Suite_for_Story_Visualization_CVPR_2026_paper.pdf

A Metric Definitions

This appendix gives the full definition, formula, and computation of every metric used in Section 5. Table 8 is the master abbreviation list; the abbreviations are used throughout the experiment tables. Unless stated otherwise, all embedding-based metrics use the multilingual sentence encoder `paraphrase-multilingual-MiniLM-L12-v2` with L_2-normalized outputs, so that a dot product equals a cosine similarity. This encoder is chosen because AI prompts are in English while human-director descriptions are in Chinese, requiring cross-lingual alignment. Every metric runs on CPU.

Continuity recall (REC). The core continuity metric. For an entity e and a shot pair $i < j$ in which e is active in both, a continuity defect $\delta(e, i, j)$=1 is recorded when $\pi_i(e)$ contradicts $\pi_j(e)$ without narrative motivation (Section 3). The LLM judge labels which defects need repair and which repairs succeed, giving Eq. (3):

$$\text{REC} = \frac{\#\{\text{defects successfully repaired}\}}{\#\{\text{defects judged in need of repair}\}} \in [0, 1].$$

Alignment layer (reference-based). AI and human-director storyboards segment the same script at different granularities, so we soft-align them before any field comparison. The per-shot visual descriptions of both sides are encoded and form a cosine similarity matrix $\text{sim} \in \mathbb{R}^{N_{\text{AI}} \times M_{\text{human}}}$; a Hungarian assignment on $-\text{sim}$ yields the optimal one-to-one matching, and pairs with similarity below $\tau_{\min}$=0.3 are discarded (CEAF-style [22]).

Soft-match P/R/F1 (SM-P/R/F1). Let Φ^* be the total similarity mass of the retained matched pairs, N_{AI} the AI shot count, and M_{human} the human-director shot count. Then

$$\text{SM-P} = \frac{\Phi^*}{N_{\text{AI}}}, \quad \text{SM-R} = \frac{\Phi^*}{M_{\text{human}}}, \quad \text{SM-F1} = \frac{2\,\text{SM-P} \cdot \text{SM-R}}{\text{SM-P} + \text{SM-R}}. \tag{9}$$

These measure how well the AI storyboard matches the director's in both semantics and shot count.

BERTScore F1 (BS-F1). For every matched description pair we compute a token-level BERTScore with the multilingual `bert-base-multilingual-cased` checkpoint and report the mean F_1 over pairs. It captures matched-pair description quality at a finer granularity than the sentence-level cosine.

Kendall's τ (Kτ). Sorting matched pairs by their AI-side index and reading off the reference-side indices gives a permutation; with C concordant and D discordant pairs over n matched shots,

$$\text{K}\tau = \frac{C - D}{n(n-1)/2} \in [-1, 1],$$

measuring how well the AI shot ordering preserves the director's ordering.

Field layer (reference-based). On the matched pairs we compare three cinematography fields, each normalized by a regex rule table into a closed label set: shot size (7 classes), camera angle (9 classes), and camera movement (12 classes, multi-label). Unmatched raw values map to *unknown* and each field additionally reports its coverage (comparable pairs / matched pairs).

Table 8: Master list of metric abbreviations. "Dir." is the improvement direction: ↑ higher is better, ↓ lower is better.

Abbr.	Full name	Dir.
Alignment layer (reference-based)		
SM-P	Soft-Match Precision	↑
SM-R	Soft-Match Recall	↑
SM-F1	Soft-Match F1	↑
BS-F1	BERTScore F1	↑
Kτ	Kendall's τ (ordering)	↑
Field layer (reference-based)		
SZ-ACC	Shot-size Accuracy	↑
SZ-F1	Shot-size macro-F1	↑
AN-ACC	Camera-angle Accuracy	↑
AN-F1	Camera-angle macro-F1	↑
MV-JAC	Movement Jaccard (multi-label)	↑
Distribution layer (reference-based)		
SZ-JS	Shot-size distribution JS div.	↓
AN-JS	Camera-angle distribution JS div.	↓
MV-JS	Movement distribution JS div.	↓
Reference-free layer		
EOD	Entity Out-Degree	↑
ACOS-M	Adjacent Cosine (mean)	↑
ACOS-N	Adjacent Cosine (min)	↑
NLI-CR	NLI Contradiction Rate	↓
NLI-MP	NLI Mean max contradiction Prob.	↓
Targeted probe (state-conditioned)		
TC-P	Targeted Contradiction Prob.	↓
TC-R	Targeted Contradiction Rate	↓
Image layer		
CREF	Character–Reference DINOv2 sim.	↑
CSELF	Character self-consistency (DreamSim dist.)	↓
SREF	Scene–Reference DINOv2 sim.	↑
SSELF	Scene self-consistency DINOv2 sim.	↑
CLIP-T	CLIP-T prompt fidelity	↑
AES	LAION Aesthetic score	↑
Double-blind judge, prompt layer		
CSC	Character-State Continuity	↑
SCON	Scene Consistency	↑
PRP	Prop Continuity	↑
CGEN	Composition Generability	↑
IFIT	Intent Fit	↑
Double-blind judge, image layer		
CAPP	Character Appearance	↑
SLAY	Scene Layout	↑
PRPI	Prop Continuity (image)	↑
SFQ	Single-Frame Quality	↑
Continuity recall (LLM-judged)		
REC	Continuity Recall	↑

Accuracy and macro-F1 (SZ/AN-ACC, SZ/AN-F1). For the single-label size and angle fields, accuracy is the fraction of matched pairs with pred=gold, and macro-F1 is the unweighted mean of per-class F_1 over classes appearing in gold or pred.

Movement Jaccard (MV-JAC). Movement is multi-label; for each matched pair with label sets A (pred) and B (gold),

$$\text{MV-JAC} = \frac{1}{|\mathcal{M}|} \sum_{(A,B) \in \mathcal{M}} \frac{|A \cap B|}{|A \cup B|} \in [0, 1].$$

Table 9: Multi-dimensional definition of shot state (d=8 dimensions), referenced from Section 4.2.1.

Dimension	Meaning
scene	physical scene / location
atmosphere	lighting, color tone, mood
characters	on-screen characters
character states	per-character emotion and posture
spatial	spatial relations / blocking of characters, objects
props	key props and possession relations
actions	in-shot actions / events
camera style	shot size / angle / movement

Distribution layer (reference-based, alignment-free). This layer compares label *distributions* without any shot alignment, sidestepping shot-count mismatch. For each field, per-episode label histograms P (AI) and Q (director) are built over the closed label space with ε=10^{-6} smoothing and normalized (multi-label movement is expanded per label). With $M = \frac{1}{2}(P+Q)$ and base-2 logarithms,

$$\text{JS} = \tfrac{1}{2}D_{\text{KL}}(P\|M) + \tfrac{1}{2}D_{\text{KL}}(Q\|M) \in [0,1],$$

computed as SZ-JS, AN-JS, MV-JS. Lower means the AI camera-language distribution is closer to the professional director's.

Reference-free layer. These metrics need no human reference and are computed on a single storyboard.

Entity out-degree (EOD). Characters and props are CSV columns, so no NER is needed. The shot–entity bipartite graph is projected onto a shot graph; for every pair $i < j$ the edge weight is the number of shared entities. With n shots,

$$\text{EOD} = \frac{1}{n}\sum_{i<j}\left|\text{ent}(i) \cap \text{ent}(j)\right|,$$

measuring cross-shot entity carry-over [8].

Adjacent cosine (ACOS-M, ACOS-N). With $\mathbf{v}_i$ the encoded description of shot i, we report the mean and min of adjacent similarities $\mathbf{v}_i \cdot \mathbf{v}_{i+1}$, measuring local coherence [46].

NLI contradiction (NLI-CR, NLI-MP). Using the multilingual NLI model `mDeBERTa-v3-base-xnli. . . -2mil7` [19], each shot h_i is scored against its previous k=3 shots as premises, and the shot-level contradiction score is the maximum:

$$c_i = \max_{i-k\le j<i} P_{\text{NLI}}(\text{contradiction} \mid h_i, h_j). \tag{10}$$

We then report NLI-MP, the episode mean of c_i, and NLI-CR, the fraction of shots with $c_i \ge 0.5$ (SummaC-style [17]).

Targeted state-conditioned contradiction probe. The targeted probe measures repair only at the sparse sites the selective filter flags. For each flagged modification we take the graph-expected entity state as the NLI premise and score the shot description as hypothesis, before and after injection. Over the flagged set we report

$$\text{TC-P} = \text{mean of } P_{\text{NLI}}(\text{contradiction} \mid \text{expected state}, d),$$
$$\text{TC-R} = \text{fraction of flagged shots with } P_{\text{NLI}} \ge 0.5,$$

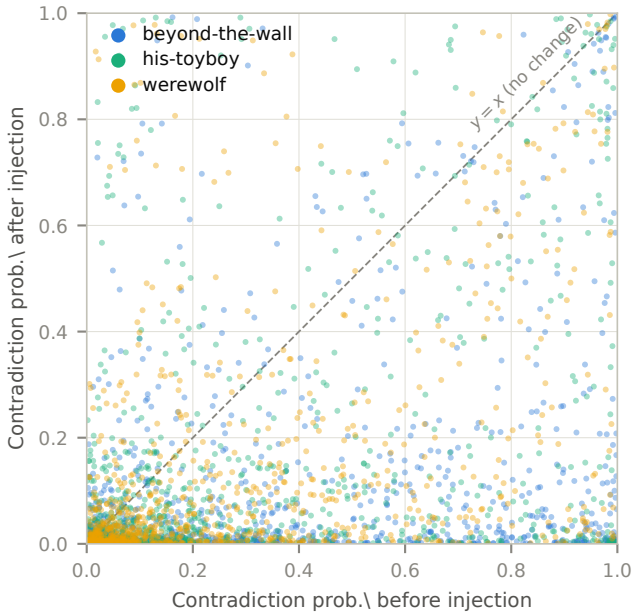

Figure 5: Targeted state-conditioned contradiction probability before vs. after memory injection for all 2,946 repair-flagged shots (three dramas × six backbones, one point per shot). Points below the y=x diagonal are repaired; the mass collapsing onto the x-axis corresponds to the two-thirds contradiction drop of Table 3, and the sparse above-diagonal points are the residual failures discussed in Section 5.4.

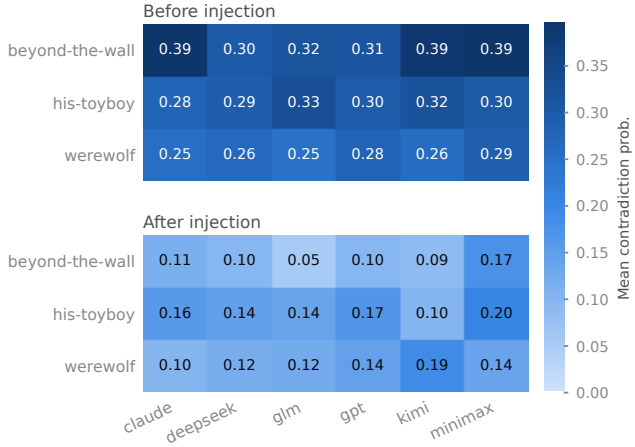

Figure 6: Mean targeted contradiction probability per drama–backbone cell, before (top) and after (bottom) memory injection. All 18 cells improve, from a 0.25–0.39 band down to 0.05–0.20 (Section 5.4).

for $d \in \{\text{before}, \text{after}\}$. Repair is effective when the *after* value drops well below the *before* value. Figure 5 plots all 2,946 shot-level before/after pairs behind Table 3.

Image layer. Keyframes are scored with open vision checkpoints on CPU, with a sha1 feature cache. Face crops use MediaPipe BlazeFace (confidence 0.5, box expanded 30%) with a full-image fallback when no face is found (detection rate 0.48). Metrics are computed

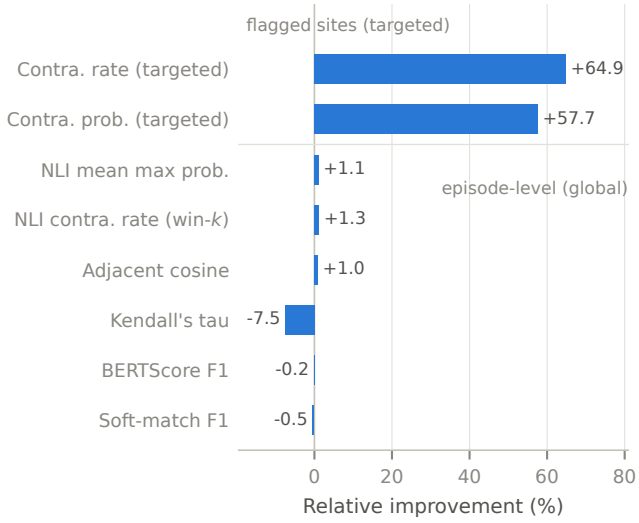

Figure 7: Relative improvement at flagged sites versus at the episode level. The two targeted metrics gain 57.7% and 64.9%; every global metric moves by at most 1.3%, the dilution effect of Section 5.4.

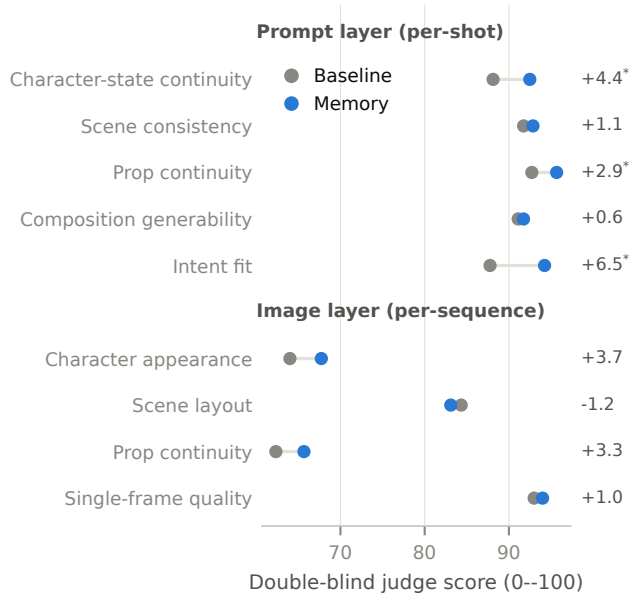

Figure 8: Double-blind judge scores by dimension (data of Table 5): baseline (gray) vs. memory (blue), pooled over six backbones; * Holm-corrected $p < 0.05$. The prompt layer shows large, significant gains on the dimensions injection edits, while image-layer deltas point the same way but are attenuated by the stochastic text-to-image stage (Q3).

only on shots that both conditions generated successfully, so safety-filter refusals never bias the pairing.

DINOv2 similarities (CREF, SREF, SSELF). With DINOv2 (`facebook/dinov2-base`) CLS embeddings, L_2-normalized: CREF is the mean cosine between a generated face crop and the character reference faces; SREF the mean full-image cosine between a generated frame

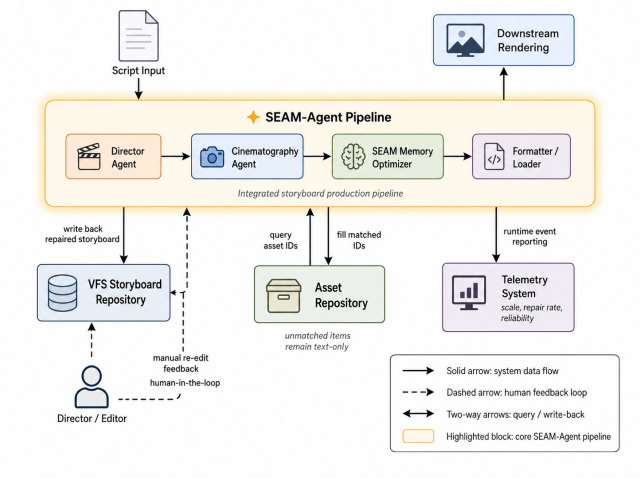

Figure 9: Online deployment of SEAM as the memory-optimization stage of the SEAM-Agent storyboarding pipeline in CreativeFitting's production system. A script flows through the Director, Cinematographer, and SEAM Memory Optimizer agents to the Formatter/Loader and on to downstream rendering. Solid arrows are system data flow; the bidirectional arrows are asset query and write-back against the shared repository; dashed arrows are the feedback from directors and editors that keeps a human in the loop; the highlighted block is the core SEAM-Agent pipeline.

and its scene reference; SSELF the mean full-image cosine between generated frames sharing a scene. All in $[-1, 1]$, higher is better.

Character self-consistency (CSELF). For frames sharing a character, we average the DreamSim distance [6] (`dino_vitb16`, distance = 1 − cos). Lower means a more stable cross-shot appearance.

CLIP-T (CLIP-T). Mean cosine between the CLIP (`clip-vit-base-patch32`) image embedding and the text embedding of the visual-description part of the keyframe prompt, measuring image–prompt fidelity.

Aesthetic (AES). The LAION improved-aesthetic linear head [32] on a CLIP ViT-L/14 embedding, yielding a 1–10 single-frame quality score.

Statistics. All baseline-vs-memory contrasts are paired at the episode level. We use the Wilcoxon signed-rank test [38] on the differences diff = memory − baseline (fewer than six nonzero pairs yields no p-value); the matched-pairs rank-biserial effect size is $r = (W_+ - W_-)/(W_+ + W_-)$; and p-values are Holm-corrected within each metric family at α=0.05 [11] (* marks corrected $p < 0.05$).

B Online Deployment Architecture

This appendix details the production pipeline behind the online run of Section 5.3. In CreativeFitting's system, SEAM is not a standalone tool but the memory-optimization stage of SEAM-Agent, a multi-agent storyboarding pipeline that turns a raw episode script into a rendering-ready shotlist. Figure 9 shows the full data flow.

A script enters the pipeline and passes through three agents in sequence. The *Director Agent* decomposes the script into shots

and fixes pacing and narrative intent; the *Cinematographer Agent* assigns shot size, camera angle, and movement, and resolves each shot's entities against the shared *Asset Repository*: it queries asset IDs and fills back the matched ones, while unmatched items remain text-only and are never forced onto a wrong asset. The *SEAM Memory Optimizer* is where SEAM runs as a mandatory stage: it extracts the multi-dimensional shot state, retrieves causally prior context from the memory graph, filters it selectively, and rewrites the affected visual descriptions, so continuity is repaired before the shotlist ever reaches a renderer. Finally the *Formatter/Loader* serializes the enriched shotlist for *Downstream Rendering*.

Two loops close the system around the directors who own the creative intent. The repaired storyboard is written back to the *VFS Storyboard Repository*, from which a director or editor may issue *manual re-edit feedback*; this path keeps a human in the loop, letting them override any stage without leaving the pipeline. Separately, the Formatter/Loader reports runtime events to a *Telemetry System* that tracks production scale, repair rate, and reliability. The acceptance figures of Section 5.3 are read directly from the annotated shotlists rather than from this channel, so they reflect the human decisions themselves and not aggregated monitoring counters (Table 7).