EVOREEL: EVOLVING HARNESS FOR LONG-HORIZON VIDEO GENERATION

Anonymous authors
Paper under double-blind review



Figure 1: ViStoryBench story 19 was selected post hoc to illustrate EvoReel's advantage over the Shown runners-up (Table 2). Boxes highlight mismatches with the brief's cast. EvoReel fully shows 7/29 shots, versus Co-Director's 2 and Anim-Director's 0.

ABSTRACT

Long-horizon video generation turns a brief into a minute-scale, multi-shot film. Since video models typically render clips under 30 seconds, it relies on agentic *harnesses* that must write a coherent script and render it with characters and places consistent across cuts. Existing harnesses, however, remain static. Across films, nothing learned carries over without fine-tuning or curation, and within a film, every shot is conditioned on references fixed in advance, although each shot needs different characters, places and earlier frames. To address these shortcomings, we propose **EvoReel**, an evolving harness for long-horizon video generation that learns from its own productions without updating any model. Specifically, EvoReel writes and renders every shot from memory and checks it with a self-judge. Across films, it distills its runs on training films into an experience memory of the revisions that made shots more specific and the content that drafts tended to lose. Within a film, role-specific agents write the screenplay from story to scene to shot over a structured story memory, with the shot writer consulting that experience as it first drafts each scene, and a self-judge scores each draft on professional screenwriting dimensions and sends weak shots back for revision. Rendering follows the same pattern: to keep characters and places consistent across shots, a multimodal visual memory supplies each shot with the reference images and earlier frames it calls for, and a second self-judge checks each rendered shot against its script and re-renders those that fall short. With these designs, EvoReel outperforms the strongest of eight baselines on ViStoryBench and held-out MovieBench by 28.6% and 84.1% in script composite and by up to

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.

20.2% in how much of the brief its films show, and ties for the highest character consistency under the proprietary renderer. We also introduce ReelBench, 104 original premises across 13 genres with checkable constraints, which tests agents beyond shot lists adapted from existing productions. Code and demos are available at https://anonymous.4open.science/r/EvoReel-D2EF.

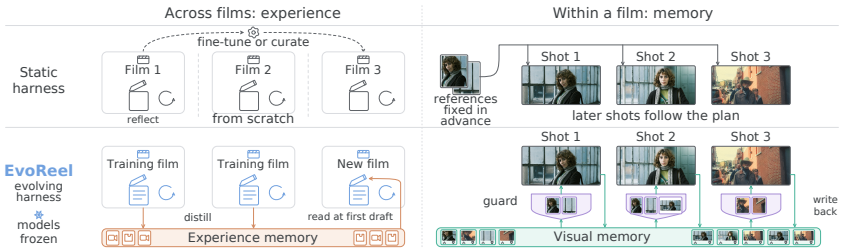

Figure 2: **Static versus evolving harnesses.** *Across films* (left), static harnesses restart unless fine-tuned or curated; EvoReel distills training films into experience memory for new scripts' first drafts, with models frozen. *Within a film* (right), static harnesses use fixed references, while EvoReel draws references from visual memory through a guard and writes each rendered shot back.

1 Introduction

Video models (Polyak et al., 2024; Team Wan et al., 2025) typically render clips under 30 seconds, so long-horizon video generation, turning a brief (a shot list or a premise) into a minute-scale, multi-shot film, relies on agentic *harnesses*: the code that decides what each model call sees and what persists across calls and films (Lin et al., 2024; Yuan et al., 2024; Wu et al., 2025b; Huang et al., 2026a). The film needs a script faithful to the brief and concrete in every shot, and clips that keep characters and places consistent across cuts.

Most harnesses hand a shot plan to a generator (Lin et al., 2024; Long et al., 2024; Wu et al., 2025b) and stay static across films and within them (Figure 2, top). Improvement rarely outlasts one sample: critique, reflection and search end with the sample (Long et al., 2026; Shi et al., 2025; Huang et al., 2026a), gains that carry across films require fine-tuning (Soni et al., 2024; Mu et al., 2026) or expert-curated experience (Wang et al., 2024a), and otherwise a general-purpose language model writes each film from scratch. Memory is equally fixed, though each shot needs different characters, places and prior frames: most harnesses generate references once before rendering (Li et al., 2024; Wu et al., 2025b; Song et al., 2026), so later shots follow the plan, not what was rendered; others relay only the previous clip's final frame (Yuan et al., 2024; Mu et al., 2026). The memories that do adapt to rendered frames (Zhang et al., 2025; Wei et al., 2026) need training or internal access to open-weight renderers, so none serves a proprietary one. Within a film, content also slips away unnoticed: rewriting a beat for the screen tends to drop part of what the brief asked for, and a critic that shares the writer's reasoning, or gives one verdict for a whole clip, cannot point to what was lost. Therefore, *how to design a harness that learns across films and adapts within a film*, without retraining any model, remains a challenge for long-horizon video generation.

To resolve this challenge, we propose EvoReel, an evolving harness that wraps frozen language, vision-language, image and video models in memories and self-judges at two levels. Across films, it distills its own runs on training films into an experience memory that guides each new script as it is first drafted. Within a film, role-specific agents write the script over a structured story memory, a multimodal visual memory supplies each shot with the references and earlier frames it calls for, and self-judges check both the script and every rendered clip. Because what EvoReel learns is kept in memory rather than in weights, it learns across films and adapts within a film without retraining any model, and so serves proprietary renderers as well as open-weight ones. Our contributions are:

- We propose EvoReel, an evolving harness for long-horizon video generation that accumulates experience across films while keeping all model weights frozen. Its training harness distills script-writing trajectories into visual-treatment exemplars, specificity-improving revisions and omission notes, which guide the shot writer in composing descriptions and visual treatments from the first draft rather than only revising flagged shots.

• We design two memories indexed by the script's characters, locations and beats. Role-specific agents compose the story, scenes and shots over a structured story memory that exposes each call only to relevant entries, while a multimodal visual memory of reference images and rendered frames retrieves the references each shot requires.

• We design script and video self-judges that localize failures instead of issuing a holistic verdict. The script self-judge, operating in a context isolated from the writers' prompts, attributes each low-scoring screenwriting dimension to the responsible shots, and the video self-judge verifies each clip against its script line at the clause level and re-renders the shots that fail.

• We construct VLLM-based video judges and ReelBench, whose 104 original open-ended premises in 13 genres and 727 checkable constraints test agents beyond adapted shot lists, closer to out-of-distribution use. Against eight baselines, EvoReel leads the script composite on all three corpora, two held out, and its films cover 12.5% and 20.2% more briefed shots on ViStoryBench and MovieBench and tie for the highest character consistency under the proprietary renderer.

2 Related Work

2.1 Agentic video generation

Large diffusion transformers, open (Polyak et al., 2024; Kong et al., 2024; Yang et al., 2025; Team Wan et al., 2025; HaCohen et al., 2025) and closed (OpenAI, 2024), dominate text-to-video generation, turning a prompt into a few seconds of video; downstream systems treat them as frozen services. A film requires splitting a minute-scale narrative into executable shot prompts and keeping identity and location consistent across separately rendered cuts. Agentic systems supply this layer: a language model plans rather than renders, and its artifact is a script. Four patterns recur: planners whose plans a generator executes (Lin et al., 2024; Long et al., 2024; Wang et al., 2026b), routers to specialised agents (Yuan et al., 2024; Wu et al., 2025b; Xie et al., 2024), critics that self-improve within one sample (Soni et al., 2024; Long et al., 2026), and collaborators keeping a human or director agent in the loop (Zhuang et al., 2024; Shi et al., 2025; Wang et al., 2026a; Zheng et al., 2024). A separate line builds cross-shot consistency into the renderer through visual memories indexed by camera pose, 3D surfels, filtered keyframes, decoupled story and shot queries, sparse attention over an untruncated cache, or character-addressable slots (Yu et al., 2025; Li et al., 2025; Zhang et al., 2025; Wei et al., 2026; Meng et al., 2026). These are trained into, or need internal access to, open-weight renderers, and StoryMem notes the failure motivating our design: without textual metadata, retrieval becomes ambiguous in multi-character scenes (Zhang et al., 2025). Evaluation likewise splits into clip-level renderer scores (Huang et al., 2024a;b; Zheng et al., 2025; Liu et al., 2024; 2023; Sun et al., 2025; He et al., 2024) and story-level benchmarks that supply an enumerated shot list (Zhuang et al., 2026; Wu et al., 2025a). Throughout, the orchestration layer is fixed once written: improvement ends with the sample, as in critique and search, or moves into fine-tuned renderer or scripter weights (Soni et al., 2024; Yuan et al., 2024; Mu et al., 2026); experience reused across tasks without retraining is rare and expert-written (Wang et al., 2024a).

2.2 Harness engineering

Outside video, much usable capability is now held to sit outside the weights, in the *harness*. Tool use added capability by changing what surrounds the model (Yao et al., 2023; Schick et al., 2023), and memory made it persist through retrievable experience and its taxonomies (Park et al., 2023; Sumers et al., 2024), self-linking notes (Xu et al., 2025) and stores of actions rather than facts (Wang et al., 2023; 2024b; Zhao et al., 2024). Recent work optimises the harness directly: AutoHarness synthesises a wrapper that filters illegal actions so a smaller model beats a larger one (Lou et al., 2026); Evo-Bench lets an agent optimise its own harness, gaining up to 16.6 points and approaching hand-engineered baselines (Huang et al., 2026b); Harness-of-Harness stacks a meta-layer over coding-agent scaffolds for a 52% relative gain (Yan et al., 2026); and APEX extracts reusable principles in a self-evolution loop (Chen et al., 2026). The benefits are consistent: capability without retraining, cheap to change and often transferable across policy models. All of these target text and code agents, and three of their properties do not carry over to video. Their actions are cheaply verifiable, whereas whether a clip renders its script line needs a model judge, so EvoReel's video self-judge asks closed questions about single clauses and its script self-judge scores drafts apart from

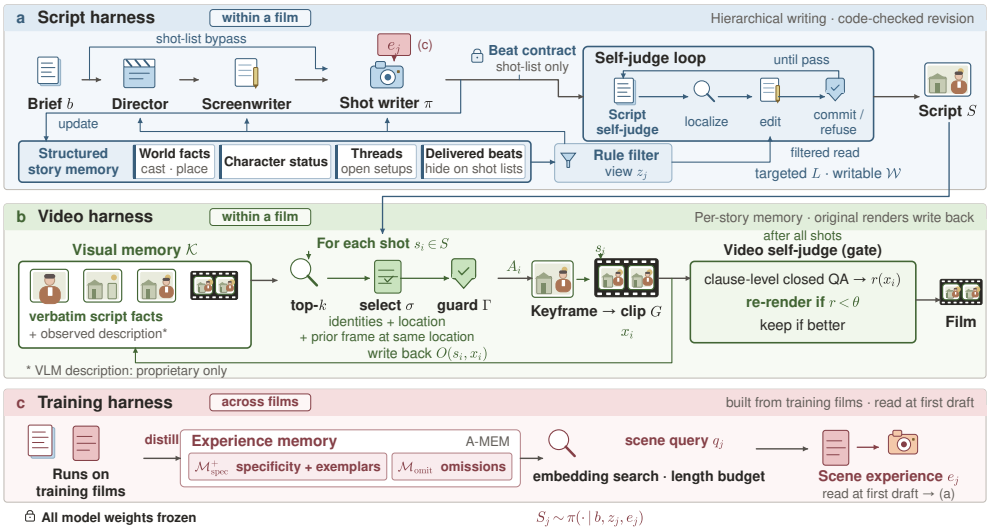

Figure 3: Overview of EvoReel; all model weights stay frozen. **(a)** Role-specific agents write the script S over a structured story memory, and a self-judge's edits are committed only within the writable fields $\mathcal{W}$. **(b)** Each shot s_i draws references from a multimodal visual memory $\mathcal{K}$ through selection σ and guard Γ and writes its frames back; a gate re-renders failing shots. **(c)** An experience memory built offline from training traces gives the shot writer π each scene's experience e_j.

the writers' reasoning. Their output is the artifact, whereas a script is a plan a frozen renderer executes; since a failure may lie in either, EvoReel gives each its own memory and self-judge. Here the persistent state is visual and long-lived, so EvoReel keys its visual memory to the script's characters and places, supplying the textual metadata StoryMem lacks (Zhang et al., 2025). Its evolution also differs: rather than rewriting its own code like AutoHarness and Evo-Bench, EvoReel evolves what its harness carries, an experience memory distilled from its own runs.

3 Methodology

EvoReel keeps its pretrained language, vision-language, image and video models frozen, so that it also serves proprietary renderers, and contributes only the harness around them. Given a brief b, it writes a script $S = (s_1, \ldots, s_n)$ whose shots have fields $\mathcal{F}$, including the action description, cast, setting and camera direction, and renders one clip per shot. For ReelBench, the brief b includes both the premise and its checkable constraints. Within a film, a script harness (Section 3.1) and a video harness (Section 3.2) each write from a memory and are checked by a self-judge; across films, a training harness (Section 3.3) supplies an experience memory. The structured story memory and the multimodal visual memory index their records by the film's characters, locations and beats, and the experience memory by the situation in which a shot was written.

3.1 Script harness

A minute-scale script has dozens of shots that must all keep the brief's content and stay consistent with each other. We split the writing across three role-specific agents. A director plans the story, a screenwriter breaks it into scenes, and a shot writer writes the shots, so a failure can be traced to its source. A shot list goes straight to the shot writer. The agents share a structured story memory of world facts, character states, open threads and delivered beats, which a language model updates after every scene. Each call reads only what a fixed filter selects by the scene's cast and location, so the writer sees the current beat without events it should not know yet. Shot lists need extra care, because they already fix what happens, where and to whom. There the filter hides delivered beats, since telling the writer not to repeat them would in effect tell it to drop briefed content. A *beat contract* also requires each shot's description to carry its beat and takes cast, setting and order from the beat, so the writer can rephrase a beat but not who is in it or where. To stop an editor from undoing this, edits are limited to writable fields $\mathcal{W}$, namely shot size, camera movement, duration

and a *visual treatment* (a note on framing, lens, lighting and blocking that may add no character or dialogue absent from the beat), plus the description on an open premise. Nothing above checks whether a draft is actually good screenwriting, so a script self-judge reviews each one. It scores the draft against the brief on D screenwriting dimensions, ranging from story structure to brief coverage. The judge runs in a fresh context and never sees the writing prompts, because a critic that reads the writer's reasoning would simply adopt its justifications. In round t, let J_d be its score on dimension d. The dimensions below a threshold τ form F_t, a localizer finds the shots L_t responsible for them, and an editor E revises those shots. A revision may change only the target shots, and only their writable fields. For target shots L, the admissible revisions of S are

$$\mathcal{A}_{\mathcal{W}}(S; L) = \big\{S' : s'_i = s_i \ \ (i \notin L), \ \ s'_i[f] = s_i[f] \ \ (i \in L, \ f \notin \mathcal{W})\big\}.$$

Each proposed revision is checked to ensure that changes are confined to the writable fields of the targeted shots. We accept the revision if it passes this check; otherwise, we retain the current script:

$$F_t = \{d : J_d(S_t, b) < \tau\}, \qquad \tilde{S}_t = E(S_t, L_t, F_t), \qquad S_{t+1} = \begin{cases} \tilde{S}_t, & \tilde{S}_t \in \mathcal{A}_{\mathcal{W}}(S_t; L_t), \\ S_t, & \text{otherwise}, \end{cases} \tag{1}$$

The current script is returned at termination. Revision acceptance enforces the editing constraint rather than requiring an increase in the judge score.

3.2 VIDEO HARNESS

A video model renders each clip separately, so a character's appearance and a location's layout must come from outside it. Since shots need different characters and places, and later shots need what earlier ones rendered, no reference set fixed in advance serves them all. EvoReel instead keeps a per-story visual memory $\mathcal{K}$, growing as the film is rendered, whose records pair an image (a character sheet, an empty location plate, a keyframe or a rendered frame) with the script's facts about it. On the proprietary renderer only, a vision-language model also describes each image, so the memory records where a shot departed from its script. For shot s_i, candidates are ranked by exact character and location overlap, text match and recency, an LLM selects among the top-ranked, and the chosen references are composed into a keyframe that is animated. The new frames are written back,

$$A_i = \Gamma\big(\sigma(\mathrm{top}_k(\mathcal{K}; s_i))\big), \qquad x_i = G(s_i, A_i), \qquad \mathcal{K} \leftarrow \mathcal{K} \cup O(s_i, x_i), \tag{2}$$

where σ is the model's selection, G the keyframe-then-video render and O the new records. Since a selection can miss what a shot needs most, the guard Γ adds, within a reference limit, the location's plate, the sheet of any character in the opening frame, and the previous shot's final frame when the location is unchanged. A conditioned clip can still miss its script line, so a video self-judge checks each clip. It avoids whole-clip verdicts, which cannot tell a repair what to fix. Instead, it decomposes the script line into m_i frame-decidable commitments, tagged as story-bearing or decoration. A vision-language model describes the clip without seeing the script, and a text-only checker returns a binary answer $a_{ik}(x)$ for each commitment k on clip x. Let $\mathcal{B}_i$ index the story-bearing commitments. We use $\mathcal{I}_i = \mathcal{B}_i$ when $\mathcal{B}_i$ is nonempty, and $\mathcal{I}_i = \{1, \ldots, m_i\}$ otherwise. The same commitments are used for both renders. For a candidate render x'_i, the commitment rate and retained clip are

$$r(x) = \frac{1}{|\mathcal{I}_i|} \sum_{k \in \mathcal{I}_i} a_{ik}(x), \qquad x_i^\star = \begin{cases} x'_i, & r(x'_i) > r(x_i), \\ x_i, & \text{otherwise}. \end{cases} \tag{3}$$

Only shots with $r(x_i) < \theta$ are re-rendered, worst first and at most $\lceil \rho n \rceil$ per film, from the original keyframe with the failed clauses named in the repair instruction. A replacement is adopted only when it improves this internal commitment rate.

3.3 TRAINING HARNESS

A role prompt can describe what a well-specified shot looks like, but not which of two revisions a judge scored higher, or which parts of a beat writers tend to lose. That knowledge exists only in trajectories, and the harnesses above discard it after each film. The training harness mines it from script-harness runs on MovieBench's training split, which shares no source movie with any evaluation task, and keeps two kinds of note. A *specificity note* records a shot the editor revised, the

Table 1: The three evaluation corpora. MovieBench is held out by source movie and ReelBench by construction. The lower block lists the factors ReelBench varies by design.

	ViStoryBench (Zhuang et al., 2026)	MovieBench (Wu et al., 2025a)	ReelBench
Brief	shot list, cast	shot list, cast	premise+constraints
Tasks	80 stories	41 scene chunks	104 premises
Shots	1,317	2,875	unspecified
Genres	53 story types, 1–7 each	6 films, 4 comedies	13, 8 premises each
Cast band	1–10 characters	4–16 characters	1–14 characters
Duration floor	none stated	none stated	150–450 s

situation it was written in, and whether the revision raised the shot's production specificity under the brief-blind rubric of Section 4.1. Only rewrites of an existing visual treatment are labelled this way. Fills of an empty treatment are kept as exemplars without a verdict, because comparing them with an empty field would only reward longer text. An *omission note* lists what a draft lost from one beat, as found by a fidelity judge that reads the brief. Together the notes form the experience memory, built in A-MEM (Xu et al., 2025) and queried by embedding similarity alone. The training harness applies experience when each scene is first drafted, guiding both what its shots retain from the brief and how they are visually realized. At construction it acts before the judge and on every field of every shot, shot-list descriptions included, whereas the editor reaches only flagged shots and writable fields. The frozen shot writer π drafts the shots S_j of scene j from the brief and the scene's view z_j of the story memory, and is now also conditioned on the notes retrieved for that scene,

$$\begin{aligned} S_j \sim \pi(\cdot \mid b, z_j) \quad &\longrightarrow \quad S_j \sim \pi(\cdot \mid b, z_j, e_j), \\ e_j = \mathrm{top}_k\big(\mathcal{M}^{+}_{\mathrm{spec}}; q_j\big) &\oplus \mathrm{top}_k\big(\mathcal{M}_{\mathrm{omit}}; q_j\big), \end{aligned} \tag{4}$$

where the query q_j describes the setting and action of the scene's beats, $\mathcal{M}^{+}_{\mathrm{spec}}$ holds the specificity notes that worked together with the exemplars, $\mathcal{M}_{\mathrm{omit}}$ holds the omission notes, and $\oplus$ joins the two under a length budget. To apply this experience at construction, the shot writer is instructed to draft each shot's description and visual treatment together, using the retrieved notes as guidance. The treatment specifies how the beat is framed, lit and staged, bringing this detail into the first draft rather than leaving it to later editing.

4 Experiments

4.1 Experimental setup

We evaluate on ViStoryBench (Zhuang et al., 2026), which guided all design decisions, and MovieBench (Wu et al., 2025a), held out by source movie. Both provide annotated shot lists adapted from existing stories and screen productions, without separately checkable requirements. We therefore build ReelBench, an exploratory benchmark held out by construction, with 104 original premises in 13 genres and 727 constraints: 519 story-specific and 208 cast-band or duration requirements. All systems receive the same premise and complete constraint list per task, but no prescribed scenes or shots, and write their own shooting scripts. We render ViStoryBench and ReelBench with Veo 3.1 Fast, and ViStoryBench and MovieBench with LTX-2.3. We compare against eight published systems: ScriptAgent (Mu et al., 2026), VideoDirectorGPT (Lin et al., 2024), Anim-Director (Li et al., 2024), VideoAgent (Soni et al., 2024), ViMax (Huang et al., 2026a), MovieAgent (Wu et al., 2025b), Mora (Yuan et al., 2024) and Co-Director (Song et al., 2026). EvoReel uses the video self-judge under Veo but not LTX-2.3; most Veo shots use references added by the guard Γ without the selection step σ.

We score each script with a GPT-5.5 judge, at least five times per task. *Production specificity* rates how concretely a script specifies staging, camera and dialogue, judged without the brief, and *content fidelity* counts what it drops or alters from the brief or, on ReelBench, which constraints it fails. Their mean on $[0, 5]$ is the script composite. For video, *Shown* is the share of briefed shots a film shows (completion) or, on ReelBench, of frame-decidable constraints it meets (conformance), judged blind to the brief, and *Identity* is whether a named character looks the same across clips, not ranked on ReelBench, whose premises name no cast. Where they apply, we also report execution, the annotated

Table 2: Script composite (*Script*, on $[0, 5]$), share of the brief shown on screen (*Shown*) and character identity across cuts (*Ident.*) for every system, with each corpus rendered by the model in its header. Shading marks the best, second and third system in each column (ViStoryBench Ident.'s top two, 0.001 apart, share the best shade), and the two ReelBench video columns, where identity is scored against each system's own cast, are unshaded diagnostics. MovieBench Ident. averages the 12 films every system scores (Section C.5). All metrics indicate better with a higher value.

	ViStoryBench (n=80) video: Veo 3.1 Fast			ReelBench (n=104) video: Veo 3.1 Fast			MovieBench (n=41) video: LTX-2.3		
Method	Script	Shown	Ident.	Script	Shown	Ident.	Script	Shown	Ident.
ScriptAgent	3.063	0.345	0.841	3.872	0.511	0.814	2.000	0.613	0.815
VideoDirectorGPT	2.987	0.501	0.858	3.794	0.486	0.791	2.112	0.487	0.834
Anim-Director	2.750	0.545	0.830	4.018	0.502	0.853	2.226	0.502	0.917
VideoAgent	2.610	0.489	0.821	3.575	0.444	0.741	1.673	0.413	0.820
ViMax	2.291	0.519	0.808	3.370	0.474	0.815	2.154	0.609	0.825
MovieAgent	2.204	0.402	0.711	3.232	0.288	0.629	1.919	0.274	0.673
Mora	1.966	0.477	0.758	3.074	0.470	0.620	1.838	0.483	0.795
Co-Director	1.533	0.560	0.697	3.376	0.392	0.589	1.646	0.507	0.757
EvoReel	3.938	0.630	0.859	4.242	0.520	0.883	4.099	0.737	0.879

characters, actions and props a shot's clips show, measured for EvoReel's Veo films with the self-judge off, and place consistency. Rates other than completion are taken over delivered shots. We compare script scores with a paired Wilcoxon test and video rates with a paired bootstrap over tasks.

4.2 Main comparison

We compare the systems on both halves of the design, what the script keeps of the brief and what the film then shows (Table 2). EvoReel has the highest script composite on every corpus, 28.6%, 84.1% and 5.6% above the strongest baseline on ViStoryBench, MovieBench and ReelBench. The lead is widest where the brief fixes each shot's content, and EvoReel leads fidelity on both shot-list corpora. It does not lead every axis (Table 4): ScriptAgent is significantly more specific on ViStoryBench; Anim-Director is significantly more faithful to ReelBench's constraints, where VideoAgent and Co-Director also score above EvoReel without a significant difference; and ReelBench script length, which each system sets for itself, may account for part of the specificity differences there. On the shot-list corpora, the lead reaches the screen. EvoReel's films show 12.5% and 20.2% more briefed shots than the strongest baseline on ViStoryBench and MovieBench, and depict 8.9% and 20.5% more of the annotated characters, actions and props (Table 5); on ViStoryBench, whose Veo films pass through the video self-judge, this rate is measured for EvoReel with the self-judge off. On ReelBench, its conformance (0.520) does not differ significantly from Anim-Director's (0.502) or ScriptAgent's (0.511), and we report this column as a diagnostic. Its films also fall below the runtime floor on 36 of 104 premises, by a median of 3.9%, against 3 to 13 for six baselines and 34 and 45 for MovieAgent and ScriptAgent; with the floor counted, Anim-Director's conformance is significantly higher than EvoReel's, which is significantly higher than ScriptAgent's and MovieAgent's. On consistency, EvoReel ties VideoDirectorGPT for the highest ViStoryBench identity under Veo 3.1 Fast. Under LTX-2.3, it has the highest ViStoryBench identity (Table 5) and is second on MovieBench to Anim-Director (0.879 against 0.917, over the 12 films every system scores). We do not rank ReelBench identity, which is scored against each system's own cast. EvoReel has the highest mean ViStoryBench text-grounded place consistency under Veo (0.974; Table 5).

4.3 Ablations

We first run the script self-judge without experience. It adds 0.086 and 0.040 points of specificity on ViStoryBench and MovieBench, on the 0–5 scale, with fidelity unchanged, and raises the ViStoryBench composite by 0.046 (Figure 4). We then enable the training harness together with a request that the writer fill each shot's visual treatment at construction. Against EvoReel with neither, the pair lifts the composite by 0.158 on ViStoryBench and 0.115 on held-out MovieBench, relative gains of 4.2% and 2.9% and, on ViStoryBench, more than three times the self-judge's gain. It improves specificity on both corpora and fidelity on ViStoryBench. On ReelBench, where we separate the two, the visual-treatment request accounts for most of the specificity gain (87%) and retrieval adds the

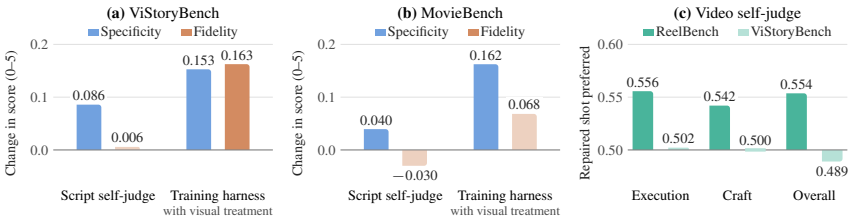

Figure 4: **(a, b)** Change in specificity and fidelity (0–5) from the script self-judge, over its first draft, and from the full training harness, over a variant that receives no retrieved experience and leaves visual treatments to the editor. **(c)** Blind LLM judges' preference for the video self-judge's repair over the original shot (0.5 = no preference), on 829 ReelBench and 76 ViStoryBench shots.

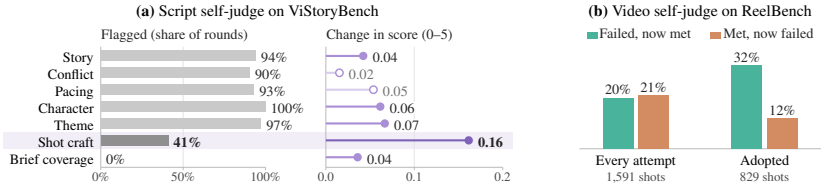

Figure 5: Where each self-judge acts. **(a)** ViStoryBench: per-dimension flag rates across rounds and draft-to-final score changes (0–5 scale). Shot craft (dark) alone is scored from an editable field; filled dots indicate significance. **(b)** ReelBench: rates of failed clauses recovered and met clauses lost on the second render, across all 1,591 attempts and 829 adopted repairs.

rest, lengthening each shot's visual treatment from 36.9 to 41.6 words and adding camera-movement language. On the video side we compare EvoReel with and without the video self-judge. With both arms rendering one script under Veo, we have a blind panel of LLM judges, Claude Opus models working in Claude Code rather than human raters, compare each adopted repair with the clip it replaced. On ReelBench, over the 829 of 6,298 pairable shots the self-judge changed, each rated by three judges, the panel prefers the repair slightly but significantly on all three questions, at 0.556 on execution, 0.542 on craft and 0.554 overall, where 0.5 means no preference. At film level the effects are small. Under LTX-2.3, where the self-judge changes between one shot in twenty and one in nine, two of the film-level columns improve significantly, MovieBench completion and ReelBench character consistency, the share of a character's scripted description that the frames show, which rises by 0.017 though its null is unmeasured. More results are in Section D.1 and Section C.4.

4.4 Component analysis

We investigate where each component acts. The script self-judge acts through the fields it may write. It raises shot craft, the dimension scored from an editable field, by 0.162, first by filling the visual treatments the draft left empty and then by refining them. On a shot list, most of its flags concern story- and scene-level dimensions that the brief already fixes, and the writable-field constraint leaves these to the brief by design, noted by the judge in 82% of its flags. Its brief-coverage scores (4.42 to 4.46) also sit above an external judge's (3.02), so fidelity is handled at construction rather than in the loop. The video self-judge concentrates on multi-character shots, flagging 83% of shots that name four or more characters against 63% of those that name one, and story-bearing and decorative clauses fail at similar rates. Because most first renders fall below θ, the repair budget sets how many shots are retried. A second render recovers about one in five failed clauses and loses about one in five met ones, and adoption keeps the render with the higher rate, so the rate rises by construction. We therefore assess repairs with the blind panel rather than with the rate, and on ViStoryBench the two measures are uncorrelated, with panel verdicts that vary with the frames shown. The training harness changes how the writer specifies shots. On ReelBench, with the visual-treatment request held fixed, retrieval lengthens the treatment from 36.9 to 41.6 words per shot and adds camera-movement language without copying the notes' wording. Three quarters of the store's notes are exemplars of filled treatments. On ReelBench, we further examine the training harness by retaining its first-draft visual-treatment instruction while withholding the retrieved notes. This instruction-only variant achieves 87% of the harness's specificity gain. Adding the notes yields the remaining improvement and increases mean treatment length from 36.9 to 41.6 words per shot.

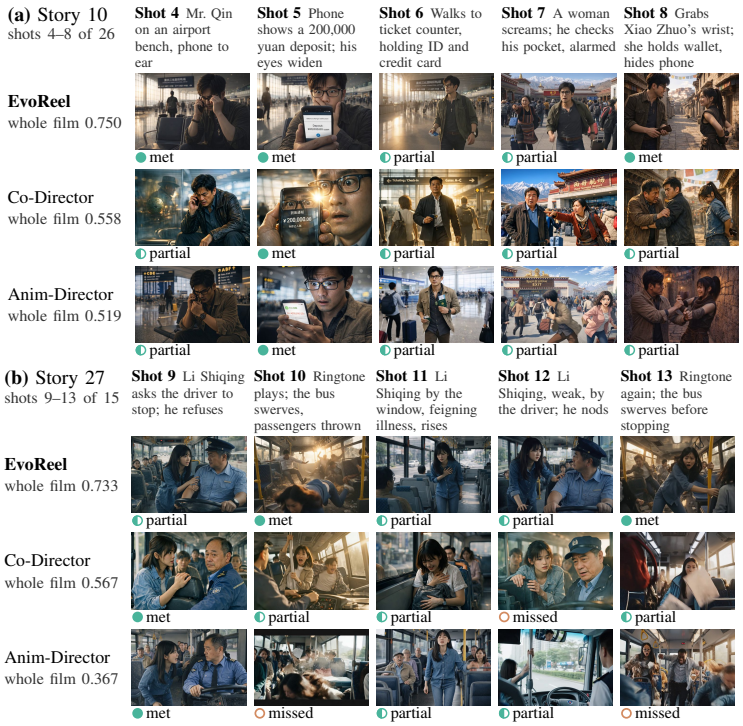

Figure 6: We compare EvoReel with the two highest-completion baselines on ViStoryBench under Veo. Columns represent brief shots; tiles show the first judge-cited clip's middle frame, with shot statuses and film-level completion reported. **(a)** Story 10 was selected from EvoReel's best-ranked films. **(b)** Story 27 was selected from those with its largest completion leads.

4.5 Frame comparison

Figure 6 illustrates completion on two ViStoryBench films. Each tile shows only the middle frame of the first judge-cited clip, not the full evidence used to assign the shot's status; partial earns half credit. EvoReel's three met shots show the airport bench, the deposit notification on Mr. Qin's phone and his wallet recovery from Xiao Zhuo in an alley. Its partials show walking along the concourse rather than approaching the ticket counter (shot 6), and running without reaching into his pocket (shot 7). Both baselines meet only shot 5. Story 27 was selected after viewing six candidates with EvoReel's largest leads over the best of Co-Director, Anim-Director and VideoDirectorGPT, excluding Story 10 and requiring all four systems to cover at least four of five consecutive shots. EvoReel scores 0.733 versus Co-Director's 0.567 and alone meets both accident shots (10, 13). The judge flags Co-Director's jacket replacing the denim shirt and Anim-Director's wrong character; EvoReel's partials concern staging. Uncited tiles show first mapped clips. Across 80 films, EvoReel's completion exceeds Co-Director's and Anim-Director's on 54 and 56 films and falls below on 16 and 18; it ties VideoDirectorGPT for the highest identity under Veo. Appendix E provides more visible results.

5 Conclusion

We presented EvoReel, a harness that learns across films and adapts within each film while keeping all models frozen. With the same video renderer used within each comparison, our results suggest that how a harness supplies context, controls edits and retains information also matters. On the shot-list corpora, the training harness's experience-guided drafting procedure yields larger script-composite gains than the script self-judge's later revisions restricted to writable fields. On open-ended briefs, blind LLM judges preferred clause-level repairs, but film-level gains remained small: improving individual shots does not necessarily improve the whole film. Future work should isolate visual memory's contribution to cross-shot consistency and calibrate automated judges against human assessments. ReelBench provides a testbed for these questions beyond adapted shot lists.

AI USE STATEMENT

We used AI assistants to help search for relevant literature, LLM/VLM as a judge, implement code, and polish the writing of this paper. The authors supervised the entire process, reviewed and verified all outputs, and take full responsibility for the content of this paper.

REPRODUCIBILITY STATEMENT

Our code is publicly available at https://anonymous.4open.science/r/EvoReel-D2EF. The method is described in Section 3 and Appendix D; the datasets, including the construction of ReelBench, in Appendix B and the evaluation protocol, metric definitions and judge prompts in Section 4.1 and Appendix C.

REFERENCES

Ya-Chuan Chen, Tien-Jen Lai, and Hsiang-Wei Hu. APEX: Adaptive principle EXtraction a three-layer self-evolution framework for production AI agents. *arXiv preprint arXiv:2606.15363*, 2026.

Yoav HaCohen, Nisan Chiprut, Benny Brazowski, Daniel Shalem, Dudu Moshe, Eitan Richardson, Eran Levin, Guy Shiran, Nir Zabari, Ori Gordon, Poriya Panet, Sapir Weissbuch, Victor Kulikov, Yaki Bitterman, Zeev Melumian, and Ofir Bibi. LTX-Video: Realtime video latent diffusion. *arXiv preprint arXiv:2501.00103*, 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 *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, 2024. doi: 10.18653/v1/2024.emnlp-main.127. URL https://aclanthology.org/2024.emnlp-main.127/.

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

Lisheng Huang, Chen Yang, Hao Zhou, Huatong Song, Zongchao Chen, Ran Le, Yang Song, Wayne Xin Zhao, and Tao Zhang. Evo-Bench: Can language models improve agent harness? *arXiv preprint arXiv:2608.09096*, 2026b.

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 *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024a.

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. *arXiv preprint arXiv:2411.13503*, 2024b.

Weijie Kong, Qi Tian, Zijian Zhang, Rox Min, et al. HunyuanVideo: A systematic framework for large video generative models. *arXiv preprint arXiv:2412.03603*, 2024.

Runjia Li, Philip Torr, Andrea Vedaldi, and Tomas Jakab. VMem: Consistent interactive video scene generation with surfel-indexed view memory. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, 2025.

Yunxin Li, Haoyuan Shi, Baotian Hu, Longyue Wang, Jiashun Zhu, Jinyi Xu, Zhen Zhao, and Min Zhang. Anim-Director: A large multimodal model powered agent for controllable animation video generation. In *ACM SIGGRAPH Asia Conference Papers*, 2024.

Han Lin, Abhay Zala, Jaemin Cho, and Mohit Bansal. VideoDirectorGPT: Consistent multi-scene video generation via LLM-guided planning. In *First Conference on Language Modeling*, 2024. URL https://openreview.net/forum?id=5PkgaUwiY0.

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 *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.

Yuanxin Liu, Lei Li, Shuhuai Ren, Rundong Gao, Shicheng Li, Sishuo Chen, Xu Sun, and Lu Hou. FETV: A benchmark for fine-grained evaluation of open-domain text-to-video generation. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

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. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2026.

Fuchen Long, Zhaofan Qiu, Ting Yao, and Tao Mei. VideoStudio: Generating consistent-content and multi-scene videos. In *European Conference on Computer Vision (ECCV)*, 2024. doi: 10.1007/978-3-031-73027-6_27.

Xinghua Lou, Miguel Lázaro-Gredilla, Antoine Dedieu, Carter Wendelken, Wolfgang Lehrach, and Kevin P. Murphy. AutoHarness: Improving LLM agents by automatically synthesizing a code harness. *arXiv preprint arXiv:2603.03329*, 2026.

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. HoloCine: Holistic generation of cinematic multi-shot long video narratives. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2026.

Chenyu Mu, Xin He, Qu Yang, Wanshun Chen, Jiadi Yao, Huang Liu, Zihao Yi, Bo Zhao, Xingyu Chen, Ruotian Ma, Fanghua Ye, Erkun Yang, Cheng Deng, Zhaopeng Tu, Xiaolong Li, and Linus. The script is all you need: An agentic framework for long-horizon dialogue-to-cinematic video generation. *arXiv preprint arXiv:2601.17737*, 2026.

OpenAI. Video generation models as world simulators. https://openai.com/research/video-generation-models-as-world-simulators, 2024. Technical report; no peer-reviewed publication.

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 *Proceedings of the ACM Symposium on User Interface Software and Technology (UIST)*, 2023.

Adam Polyak, Amit Zohar, Andrew Brown, et al. Movie gen: A cast of media foundation models. *arXiv preprint arXiv:2410.13720*, 2024.

Timo Schick, Jane Dwivedi-Yu, Roberto Dessì, Roberta Raileanu, Maria Lomeli, Luke Zettlemoyer, Nicola Cancedda, and Thomas Scialom. Toolformer: Language models can teach themselves to use tools. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

Haoyuan Shi, Yunxin Li, Xinyu Chen, Longyue Wang, Baotian Hu, and Min Zhang. AniMaker: Multi-agent animated storytelling with MCTS-driven clip generation. In *ACM SIGGRAPH Asia Conference Papers*, 2025. doi: 10.1145/3757377.3764009.

Yale Song, Yiwen Song, Nick Losier, Nathan Hodson, Ye Jin, Rhyard Zhu, Yan Xu, Daniel Vlasic, Carina Claassen, Jasmine Leon, Khanh G. LeViet, Zack Chomyn, Joe Timmons, Brett Slatkin, Scott Penberthy, and Tomas Pfister. Co-Director: Agentic generative video storytelling. *arXiv preprint arXiv:2604.24842*, 2026.

Achint Soni, Sreyas Venkataraman, Abhranil Chandra, Sebastian Fischmeister, Percy Liang, Bo Dai, and Sherry Yang. VideoAgent: Self-improving video generation for embodied planning. *arXiv preprint arXiv:2410.10076*, 2024.

Theodore R. Sumers, Shunyu Yao, Karthik Narasimhan, and Thomas L. Griffiths. Cognitive architectures for language agents. *Transactions on Machine Learning Research (TMLR)*, 2024.

Kaiyue Sun, Kaiyi Huang, Xian Liu, Yue Wu, Zihan Xu, Zhenguo Li, and Xihui Liu. T2V-CompBench: A comprehensive benchmark for compositional text-to-video generation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025.

Team Wan, Ang Wang, Baole Ai, Bin Wen, et al. Wan: Open and advanced large-scale video generative models. *arXiv preprint arXiv:2503.20314*, 2025.

Guanzhi Wang, Yuqi Xie, Yunfan Jiang, Ajay Mandlekar, Chaowei Xiao, Yuke Zhu, Linxi Fan, and Anima Anandkumar. Voyager: An open-ended embodied agent with large language models. *arXiv preprint arXiv:2305.16291*, 2023.

Jiuniu Wang, Zehua Du, Yuyuan Zhao, Bo Yuan, Kexiang Wang, Jian Liang, Yaxi Zhao, Yihen Lu, Gengliang Li, Junlong Gao, Xin Tu, and Zhenyu Guo. AesopAgent: Agent-driven evolutionary system on story-to-video production. *arXiv preprint arXiv:2403.07952*, 2024a.

Qian Wang, Ziqi Huang, Ruoxi Jia, Paul Debevec, and Ning Yu. MAViS: A multi-agent framework for long-sequence video storytelling. In *Proceedings of the 19th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2026a. URL https://aclanthology.org/2026.eacl-long.101/.

Zora Zhiruo Wang, Jiayuan Mao, Daniel Fried, and Graham Neubig. Agent workflow memory. *arXiv preprint arXiv:2409.07429*, 2024b.

Zun Wang, Jialu Li, Han Lin, Jaehong Yoon, and Mohit Bansal. DreamRunner: Fine-grained compositional story-to-video generation with retrieval-augmented motion adaptation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2026b.

Xuan Wei, Longbin Ji, Guan Wang, Xiangrui Liu, Zhenyu Zhang, Shuohuan Wang, Yu Sun, and Qingqi Hong. Memento: Reconstruct to remember for consistent long video generation. *arXiv preprint arXiv:2606.14667*, 2026.

Weijia Wu, Mingyu Liu, Zeyu Zhu, Xi Xia, Haoen Feng, Wen Wang, Kevin Qinghong Lin, Chunhua Shen, and Mike Zheng Shou. MovieBench: A hierarchical movie level dataset for long video generation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025a.

Weijia Wu, Zeyu Zhu, and Mike Zheng Shou. Automated movie generation via multi-agent CoT planning. *arXiv preprint arXiv:2503.07314*, 2025b.

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

Wujiang Xu, Zujie Liang, Kai Mei, Hang Gao, Juntao Tan, and Yongfeng Zhang. A-MEM: Agentic memory for LLM agents. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2025.

Haoyang Yan, Min le Su, Hangfan Zhang, Zhanhao Li, Chen Zhang, Shao Zhang, Yang Chen, Lei Bai, and Shuyue Hu. Harness-of-harness: Multi-day autonomous software development with continual improvement. *arXiv preprint arXiv:2609.01481*, 2026.

Zhuoyi Yang, Jiayan Teng, Wendi Zheng, Ming Ding, Shiyu Huang, Jiazheng Xu, Yuanming Yang, Wenyi Hong, Xiaohan Zhang, Guanyu Feng, Da Yin, Yuxuan Zhang, Weihan Wang, Yean Cheng, Bin Xu, Xiaotao Gu, Yuxiao Dong, and Jie Tang. CogVideoX: Text-to-video diffusion models with an expert transformer. In *International Conference on Learning Representations (ICLR)*, 2025.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. ReAct: Synergizing reasoning and acting in language models. In *International Conference on Learning Representations (ICLR)*, 2023.

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

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

Kaiwen Zhang, Liming Jiang, Angtian Wang, Jacob Zhiyuan Fang, Tiancheng Zhi, Qing Yan, Hao Kang, Xin Lu, and Xingang Pan. StoryMem: Multi-shot long video storytelling with memory. *arXiv preprint arXiv:2512.19539*, 2025.

Andrew Zhao, Daniel Huang, Quentin Xu, Matthieu Lin, Yong-Jin Liu, and Gao Huang. ExpeL: LLM agents are experiential learners. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2024.

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. *arXiv preprint arXiv:2503.21755*, 2025.

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

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. ViStoryBench: Comprehensive benchmark suite for story visualization. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2026.

Shaobin Zhuang, Kunchang Li, Xinyuan Chen, Yaohui Wang, Ziwei Liu, Yu Qiao, and Yali Wang. Vlogger: Make your dream a vlog. *arXiv preprint arXiv:2401.09414*, 2024.

CONTENTS

1 Introduction **2**

2 Related Work **3**

2.1 Agentic video generation . . . 3

2.2 Harness engineering . . . 3

3 Methodology **4**

3.1 Script harness . . . 4

3.2 Video harness . . . 5

3.3 Training harness . . . 5

4 Experiments **6**

4.1 Experimental setup . . . 6

4.2 Main comparison . . . 7

4.3 Ablations . . . 7

4.4 Component analysis . . . 8

4.5 Frame comparison . . . 9

5 Conclusion **9**

A Preliminary Human Preference Evaluation **16**

A.1 Evaluation Setup . . . 16

A.2 Results . . . 16

B Datasets **16**

C Evaluation protocol **18**

C.1 Content fidelity . . . 18

C.2 Production specificity . . . 19

C.3 Script composite and its two axes . . . 20

C.4 Video judges . . . 20

C.5 Further video columns and coverage . . . 23

D EvoReel in detail **25**

D.1 Script-side loop . . . 25

D.2 Video self-judge . . . 26

D.3 Experience memory . . . 26

E Further frame and script comparisons **27**

E.1 Script examples . . . 27

A Preliminary Human Preference Evaluation

We conducted a preliminary pairwise human evaluation of overall preference for EvoReel's generated scripts and rendered videos relative to selected baselines. The study complements the automated evaluation with holistic assessments of the final outputs.

A.1 Evaluation Setup

We randomly sampled a fixed set of 100 generation pairs (50 from ReelBench, 50 from ViStoryBench) using a fixed random seed. To ensure a rigorous comparison, EvoReel was paired exclusively against the top-performing baselines from Table 2: Anim-Director, ScriptAgent, and VideoDirectorGPT. All videos in this evaluation were rendered using Veo 3.1 Fast. The evaluation was conducted by three annotators (one author and two volunteers). The left/right presentation order of the pairs was randomized, and system names were strictly hidden from the interface. While the systems were anonymized, we acknowledge the limitation that the author-annotator might occasionally recognize stylistic artifacts specific to our own system's generations.

Evaluation Questions. Annotators were provided with the original brief and asked to choose System A, System B, or a Tie for two specific overall questions:

1. **Overall Script Preference:** *'Based on the brief, which script provides a better, more cohesive cinematic narrative?'*
2. **Overall Video Preference:** *"Which generated video sequence is better overall in terms of visual quality, consistency, and showing the briefed actions?"*

A.2 Results

We aggregate the preferences across all 300 ratings (100 pairs $\times$ 3 annotators). Win rates are calculated by assigning 1 point for a win and 0.5 points for a tie. To ensure statistical rigor, we report the 95% Confidence Intervals (CI) computed via a pair-level bootstrap (4000 resamples), alongside the Fleiss' κ (calculated across the three-class outcomes: win, tie, loss) to measure inter-rater reliability.

Table 3: Human pairwise preference for EvoReel against three selected baselines, with 300 ratings per question (100 pairs $\times$ 3 annotators). Ties count as half a win; 95% confidence intervals are computed by resampling pairs.

Question	EvoReel Win	Baseline Win	Tie	Win Rate	95% CI	Fleiss' κ
Overall Script	205	80	15	0.708	[0.635, 0.778]	0.65
Overall Video	188	94	18	0.657	[0.582, 0.725]	0.61

Discussion. Across the sampled comparisons, annotators preferred EvoReel for both scripts and rendered videos, with both 95% confidence intervals above 0.5 (Table 3). These pooled results provide preliminary human support for the overall quality of EvoReel's outputs relative to the selected baselines. The study assesses holistic preference, while Fleiss' κ measures agreement among human annotators; neither measures agreement with individual automated metrics.

B Datasets

Table 1 gives the size and role of each corpus. On both shot-list corpora the brief is text alone: a cast list with a one-line description of each named character, followed by a numbered shot list whose entries give the setting, the action, occasionally with a quoted line, the camera framing and the characters on screen. These annotations are what fidelity and execution are counted against. ViStoryBench (Zhuang et al., 2026) contributes the 80 stories of its full configuration, which span 53 labelled story types, from fairy tales, folk tales and legends of many traditions to picture books, science fiction and romance, with 4 to 30 shots (median 16) and 1 to 10 named characters (median 4) per story. The benchmark defines no training or held-out split, and every design choice in EvoReel and in the specificity rubric was made with all 80 stories visible, so ViStoryBench serves as the

development set and MovieBench and ReelBench supply the held-out comparisons. No note in the experience memory is therefore derived from it.

MovieBench (Wu et al., 2025a) annotates films hierarchically by scene and shot, and its test split holds six: *Juno*, *Bad Santa*, *This Is 40*, *The Ugly Truth*, *Harry Potter and the Prisoner of Azkaban* and *Les Misérables*. At 276 to 656 annotated shots a film is too long to plan in one response, so we keep all 2,875 annotated shots and 846 scenes and cut each film, in order, into contiguous chunks of 60 to 80 shots that never split a scene, which gives 4, 8, 8, 5, 9 and 7 chunks for the six films; the chunk is our unit, not one that MovieBench defines. Before the shot list, a chunk's brief states its position in the film and the shot range of each of its scenes, so every system receives the scene grouping. A chunk averages 70 shots and names 4 to 16 characters (median 8), and because the annotations cover a film sparsely, its 4.7 minutes of annotated footage span about 16.5 minutes of the source film. Each chunk is generated as a separate task, so MovieBench measures planning within a chunk rather than across a whole film. The training harness (Section 3.3) draws on MovieBench's training split, cut by the same rule into 368 chunks from 42 films after excluding 43 films whose annotations are incomplete or cannot be partitioned this way, and the 4,846 notes of the experience memory come from 293 of these chunks, spread over all 42 films. None of the 42 is a test film; three are other *Harry Potter* films that share the principal cast of *Prisoner of Azkaban* and supply 597 of the notes, so for that film's nine chunks the split is by film and not by cast.

ReelBench, which we built for this study, supplies what the shot-list corpora lack: briefs that leave the plan to the system, and requirements that can be checked without a shot list. Its structure is fixed by a design grid before any prose is written. The grid crosses 13 genres (drama, romance, comedy, action, crime, mystery, horror, science fiction, fantasy, historical, political intrigue, animation and short-form drama, eight premises each) with five duration floors from 150 to 450 seconds in steps of 75, held by 13, 13, 26, 26 and 26 premises, and with five bands for the number of named recurring characters, 1–2, 3–4, 5–7, 8–10 and 11–14, held by 21 premises each except 20 for the largest. A band is a required range rather than a cap, so cast size is a property of the task and not a choice left to the system, and within each genre five premises pair the five floors one-to-one with the five bands, so that each genre alone spans every level of both factors. Each premise's number of story-specific constraints is drawn from a normal distribution with mean 5 and standard deviation 2, truncated to 1 to 9, and over the 104 premises floor, cast band and constraint count correlate pairwise by at most 0.09 in magnitude. Each premise is then written against its grid cell in 57 to 88 words that set up a situation and say what the audience must come to understand, without naming a character the film must contain and without a shot list or an enumerated beat, and it closes with a sentence such as "Carry between 1 and 2 named recurring characters; at least 150s." Automated checks enforce these properties and reject any two premises whose word-trigram Jaccard overlap exceeds 0.30.

Of the 727 checkable constraints, the 519 story-specific ones are behavioural requirements that a reader can mark met, partly met or violated from a script, and at least one per premise encodes the capability the premise was written to test. For the premise that closes with the sentence above, in which a locksmith returns an object to a deposit box he once fitted so that the audience infers an earlier crime it never sees, they include that the object is established as connected to him before it is returned and that the film never depicts or narrates the original theft. Abstract qualities such as coherence are left out, so that every constraint names something a reader can locate in the script. The remaining 208 constraints specify the cast band and duration floor, which are also stated verbatim in each premise. All baselines and EvoReel receive the same premise and complete constraint list for each task before generation. We additionally audit how much of the constraint list is implied by the premise alone. Of the 89 story-specific constraints that state a number, 60 specify one absent from the premise. We audit all 519 story-specific constraints with GPT-5.5, asking whether a competent screenwriter given only the premise would know that each requirement was expected and instructing it to label uncertain cases as derivable. The audit identifies 404 as derivable. These labels characterize the information conveyed by the premise; they do not determine the constraints supplied to the systems. Script fidelity evaluates all 727 constraints, whereas the reported video-conformance diagnostic is restricted to the 404 premise-derivable story-specific constraints (Section C.4).

C EVALUATION PROTOCOL

The two fidelity rubrics and the specificity prompt are run with GPT-5.5 as judge. Prompts are quoted as run, including their em dashes and double hyphens.

C.1 CONTENT FIDELITY

ReelBench provides a premise and an explicit constraint list rather than a shot list, so fidelity is evaluated constraint by constraint. The evaluation uses the complete list supplied to every system, covering all 727 requirements across the benchmark. The judge returns scores for all four dimensions in one call. We use its dimension-3 score as fidelity; the revised anchor asks the judge to count missing or altered elements beat by beat and map those counts to a score.

> You are a script supervisor scoring a shooting script against the BEAT SHEET it was told to render. The brief you are given is not background context — it is a contract. Apply the anchors strictly and quote concrete evidence from the script before each score.
> THE CONTRACT, and it is asymmetric:
> **ADDING is ALLOWED and is NEVER penalised.** Extra visual detail, extra camera language, extra atmosphere, extra shots, invented props or background business are all fine. Never deduct for richness the brief did not ask for, and never reward it either — it is simply neutral.
> **REMOVING is penalised.** Anything a beat states — its setting, characters, action, stated expression, and its dialogue or narration line — must be rendered somewhere in the script, in any wording. Paraphrase is fine; splitting one beat across several shots is fine; omitting it is not.
> **ALTERING is penalised, and more heavily than adding is rewarded (which is not at all).** Replacing a stated action, emotion or position with a different one is a failure even when the replacement is better cinema.
> Judge fidelity to the contract, not how cinematic the prose reads.

> *3) Content Fidelity (0-5) – does the script render EVERYTHING each beat states?*
> METHOD (follow it; do not score on overall impression): go beat by beat. For each beat list the elements it states – setting, characters, action, stated expression, dialogue/narration – and mark each KEPT, MISSING, or ALTERED. Do NOT judge the camera setup here; shot size and angle belong to dimension 2. Then COUNT the MISSING and ALTERED across the whole script and read the score off that count.
> A noticed omission MUST lower the score. If your evidence mentions that something was dropped, softened, or replaced, the score cannot be 5. Additions are neutral: never raise the score for material the brief did not ask for, and never lower it either.
> State the counts in your evidence, e.g. "2 MISSING, 1 ALTERED".
> 5 = 0 MISSING and 0 ALTERED. 4 = 1-2 elements MISSING or ALTERED in total. 3 = 3-5 elements MISSING or ALTERED. 2 = 6-10 elements MISSING or ALTERED. 1 = more than 10, or several beats lose their core content. 0 = most beats gutted.

The judge states its count in 48 of 640 scored rationales, and in 45 of these the score matches the count.

A ReelBench premise states no beats, so the countable unit is the task's list of checkable constraints; the premise brief that every system writes from states two of its items, the cast band and the duration floor. The judge sees the numbered list, not the premise, and scores this single dimension. The counting mirrors the shot-list rubric, but its asymmetry does not: exceeding a requirement is neutral only where the requirement leaves room, and one that states an upper bound or an exact range is failed by exceeding it.

> **EXCEEDING AN OPEN REQUIREMENT is ALLOWED and is NEVER rewarded.** Where a requirement sets a floor or leaves room ('at least 150 seconds', 'the reveal is a distinct beat'), a script may invent business, add texture, or run longer than asked. Do not raise the score for ambition and do not lower it for it either — it is neutral.
> **BUT a requirement that states an UPPER BOUND or an exact range is NOT open, and exceeding it is a VIOLATION like any other.** 'Between 3 and 4 named recurring characters — not fewer, and not more' is failed by a script with five as surely as by one with two. Count them and say the number. This is the one place where more is worse, and it is worse because the requirement said so, not because the addition was bad.

> *1) Constraint Satisfaction (0-5) – does the script satisfy every hard requirement the task states?*
> METHOD (follow it; do not score on overall impression): take the requirements one at a time, in the order given. For each, quote the part of the script that satisfies it, or say that nothing does, and mark it MET, PARTIAL, or VIOLATED. MET: the script clearly does what the requirement asks. PARTIAL: the script gestures at it but does not carry it – e.g. a requirement to make a change legible through staging is answered by stating the change in dialogue. VIOLATED: the script does the opposite, or does not attempt it at all. A requirement the script simply ignores is VIOLATED, not PARTIAL. Then COUNT and read the score off the count.
> 5 = every requirement MET. 4 = exactly one PARTIAL and none VIOLATED. 3 = one VIOLATED, or two to three PARTIAL. 2 = two VIOLATED, or more than half PARTIAL. 1 = three or more VIOLATED. 0 = most requirements violated, or the script is not attempting this task.

The bands are stated mainly in counts rather than fractions, so that a violation costs the same on every task however many constraints it carries.

C.2 Production specificity

The scorer and its prompt call this axis craft. It asks what a script gives a crew to shoot and is judged without the brief, for the reason given at the end of the first block below. After its first line, the prompt reproduces the 0–5 anchor lines of all four dimensions of the rubric published with ScriptAgent, keeping only one dimension header among them, that of dimension 3, *3) Content Completeness (0-5) – rich, actionable filming detail; supplies visual info beyond the bare premise?*; it then fixes two written endpoints:

> You are a script supervisor. Score ONE dimension of this shooting script.
> Judge only what this text itself gives a crew to shoot: how specific and complete its visual, spatial, staging and camera information is, on its own terms.
> USE THE WHOLE SCALE. Measured over 150 scorings, 116 came back as exactly 4.0 and the entire range used was 3.0-4.2, which makes this dimension almost useless. Calibrate against these two endpoints and place the script between them:
> *1 or below* — a bare list of story beats. What happens and where, and nothing a camera department could act on: no framing, no lens, no lighting, no blocking, no composition, and no line worth speaking. However complete and well-ordered the STORY is, a text that only narrates events belongs here.
> *5* — every shot carries framing and shot size, camera position or movement, lighting or palette, blocking within the frame, the specific physical detail the crew needs, and – where characters speak – lines that fit them and do dramatic work. A director could shoot it without asking a question.
> If most shots have story text but only a minority carry any craft direction, that is a 2, not a 4. Do not cluster at 4.
> You are NOT given the brief. Faithfulness is scored by a separate pass; showing the brief here was measured to destroy the distinction, a verbatim beat sheet and a dressed shooting script both landing on 4.000.

The prompt then gives a rule for dialogue:

> Dialogue counts toward this score, under the following gate. Read it before scoring.
> Does the script contain spoken dialogue?
> **YES** — judge the lines. Lines that fit each character's identity and situation AND do dramatic work (advance, pressure, reveal, or complicate) RAISE the score, on the same footing as good camera or lighting work. Flat, interchangeable, or purely expository lines that only restate what the picture already shows pull

> it DOWN.
> **NO** — a shooting script that renders a story with not one spoken line is missing an entire craft channel, and that is a real deficit, not a neutral stylistic choice. Score it DOWN. The only way to recover is to demonstrate, beat by beat, that the picture SPECIFICALLY carries what speech would have carried: the information a viewer needs, and the emotional or moral turn of each beat. "The visuals are rich" does not qualify — point to the specific shots that do this work. A script that is silent and does not demonstrably do this cannot score above 3.
>
> Either way, also judge: is the information a viewer needs actually delivered somewhere, and do the key beats carry meaning beyond their literal action?

The composite uses the specificity score before the scorer's penalty for lost content, which the fidelity axis already charges.

C.3 Script composite and its two axes

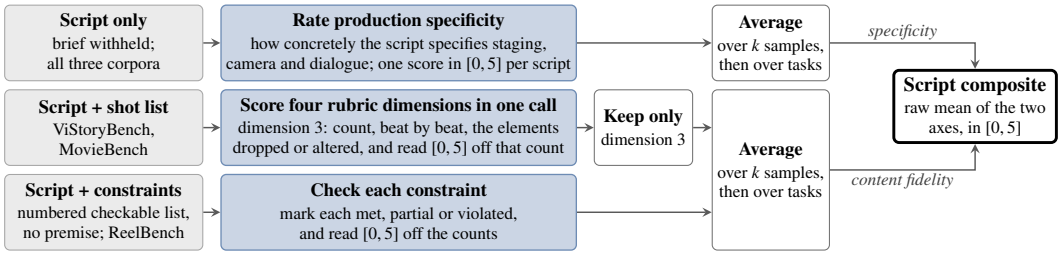

Figure 7: The script composite averages two axes on $[0, 5]$: production specificity, how concretely a script specifies staging, camera and dialogue, and content fidelity, which falls as the script drops or alters more of the brief's beat elements or, on ReelBench, violates or only partly meets more of its checkable constraints. Grey boxes are inputs, blue boxes model calls, white boxes steps in code, and the bold box the reported metric.

Both axes enter the composite on their native 0–5 scales and are averaged raw, since a z-score would depend on which systems are in the pool. They measure different things, correlating at only $r = 0.192$ over 1,199 task-system cells, and a specificity judge that never sees the brief cannot notice a script about the wrong story. The composite was fixed after the n=10 pilot and the specificity scores became available, but before the n=80 fidelity scores of EvoReel's final scripts existed. After a first composite result, the specificity prompt was revised: an earlier dialogue rule, under which writing any dialogue lowered the score, was replaced by the rule quoted above; every specificity score reported here uses the revised prompt.

Table 4 separates the script composite of Table 2 into its two axes. Under the paired Wilcoxon test of Section 4.1, EvoReel's lead over the next system is significant on both MovieBench axes, on ViStoryBench fidelity and on ReelBench specificity, and so are ScriptAgent's lead in ViStoryBench specificity, where it is ahead on 54 of the 80 tasks, and Anim-Director's lead in ReelBench fidelity. EvoReel's ReelBench fidelity does not differ significantly from VideoAgent's or Co-Director's, and its ReelBench composite lead over Anim-Director is significant. In both tables the baselines' ViStoryBench fidelity averages eight judge samples per task where every other cell averages five, with judge and rubric unchanged, and Mora's covers 79 of the 80 tasks.

C.4 Video judges

Of the six video rates in Section 4, five are computed in code from categorical judgements of single facts; pairwise identity, which asks for a rating from 1 to 5, is the exception. This follows a pilot in which two judge models, choosing labels for the causal link and the state transition between 169 pairs of adjacent ReelBench shots from blind clip descriptions, agreed with each other at κ of 0.232 and 0.271, while one of them, asked again on 52 pairs, agreed with itself at 1.00 and 0.80. The pilot's labels differ from the facts judged for the reported columns; it motivates the categorical design.

Completion runs in three stages so that no call sees both the brief and the pixels. The first describes each clip from its frames with the brief withheld, and a request containing the brief's distinctive

Table 4: The two axes of the script composite in Table 2, each on $[0, 5]$: production specificity, how concretely a script specifies staging, camera and dialogue, judged without the brief, and content fidelity, how much of the brief it keeps (on ReelBench, conformance to the checkable constraints). The composite is their mean. Shading marks the best, second and third system in each column.

	ViStoryBench (n=80)			MovieBench (n=41)			ReelBench (n=104)		
Method	Spec.	Fid.	Comp.	Spec.	Fid.	Comp.	Spec.	Fid.	Comp.
ScriptAgent	3.995	2.132	3.063	2.930	1.070	2.000	4.053	3.690	3.872
VideoDirectorGPT	3.336	2.638	2.987	2.722	1.502	2.112	4.138	3.451	3.794
Anim-Director	3.236	2.264	2.750	2.656	1.795	2.226	3.818	4.217	4.018
VideoAgent	3.004	2.216	2.610	2.186	1.161	1.673	3.070	4.081	3.575
ViMax	3.365	1.217	2.291	2.985	1.322	2.154	4.209	2.531	3.370
MovieAgent	3.369	1.039	2.204	3.037	0.800	1.919	3.841	2.624	3.232
Mora	2.912	1.021	1.966	2.656	1.020	1.838	3.573	2.575	3.074
Co-Director	2.010	1.055	1.533	2.112	1.180	1.646	2.811	3.942	3.376
EvoReel	3.671	4.205	3.938	3.290	4.907	4.099	4.550	3.933	4.242

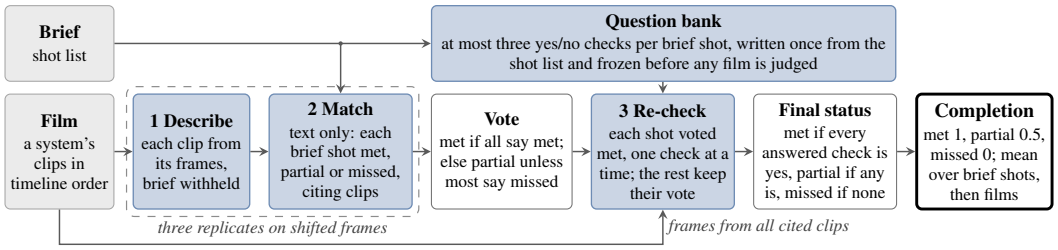

Figure 8: Completion is the share of the brief's shots a film shows, a partly shown shot counting half; no model call sees both the brief and the frames, and a shot is met only if every replicate calls it met and it passes frozen yes/no checks using frames from all clips cited as evidence for that shot. Shading as in Figure 7.

words is refused rather than sent. The second sees every clip description in timeline order with the brief's shot list and no frames, and gives each shot one status and the clips it cites as evidence:

> met — a clip's description contains the shot's action, its participants, and its setting.
> partial — a clip's description contains part of the shot but is missing the action, a participant, or the setting.
> missed — no clip's description contains the shot.
> Cite the clip numbers that are your evidence. Cite as few as the evidence supports. If you cite more than one clip you are saying the shot is spread across them, not that you are unsure which one it is. A shot you mark missed cites nothing. Do not assume shot N is clip N. Judge from the descriptions.

The first two stages run three times on phase-shifted frame grids, and a film with too few frames for disjoint grids is recorded as one effective replicate rather than three identical votes. A shot is carried forward as met only if every replicate calls it met; otherwise met votes count as partial and the majority of partial against missed decides, a tie going to partial. The third stage re-checks each shot carried forward as *met*, one closed question at a time, using sampled frames from all clips cited as evidence for that shot. Each call receives the question and these frames, but no brief text, frames from uncited clips, or system identity:

> Answer yes only if the frames clearly show it. Answer no if they do not, if they show something similar but different, or if you cannot tell. Being unsure is a no.

The questions come from a bank written once per task from the brief's shot list, at most three per shot, and frozen before any film is judged, so every system answers the same checks. A shot stays

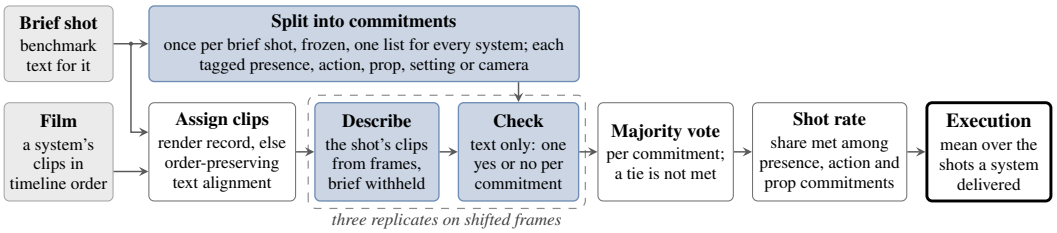

Figure 9: Execution is the share of a brief shot's presence, action and prop commitments that its assigned clips are judged to show, each decided by majority vote over replicates from descriptions written without the brief, and averaged over the shots a system delivered (under the proprietary renderer, within each film and then over films). Shading as in Figure 7.

met only if every answered check is yes, falls to partial if at least one is, and to missed if none is; a check that fails to return is dropped.

Execution applies the same describe-then-check construction to one shot at a time, against the benchmark's text for that shot rather than the script a system wrote, so every system is measured against the same target. Commitments are extracted once per brief shot and frozen, each tagged as presence, action, prop, setting or camera. The rate counts the first three, which concern the characters, their actions and their props; counting setting and camera as well would fail a clip that performs every scripted action under the wrong light. The clip is described blind, one text-only call answers yes or no per commitment, and three replicates over different frame offsets vote, a tie counting as not met. Each clip is assigned to the brief shot it was rendered for, from the system's render record or otherwise by an order-preserving text alignment (Section C.5). All clips of one shot are judged together on eight frames.

Consistency and *pairwise identity* draw two frames from two different clips of one film, chosen by normalised position within that system's own render, up to six pairs per film for consistency and sixteen for identity under the proprietary renderer, and 32 for identity on open-weight MovieBench. Every system is thus asked about the same relative positions, but a film yields pairs only where a place or character recurs across its clips, so the number of pairs differs by system (Section C.5). The *consistency panel* scores each frame separately against a text description and is told not to compare the two frames. Its scene axis, the consistency column of Table 5, takes its description of the place from the frame's own shot beat when that beat describes it, otherwise from the brief's description of the location, and otherwise from the location's first appearance in that system's own render. Only the first two sources are shared across systems, and in the runs behind that column every frame of every system drew on the beat of the shot its clip is assigned to. Its character axis, the character consistency of Section 4.3, uses a character's description, taken on ReelBench from one arm's script, so it is read only within a pair of arms that render that script. On either axis the description is split into the separate things it names, and each receives one of four verdicts in each frame:

> present — you can see it in this frame
> absent — this frame's view covers where it would be, and it is not there
> contradicted — something is there in its place that the entry rules out — a different colour, a different object, a different structure
> occluded — this frame's view does not cover where it would be: the camera faces elsewhere, or it is behind something, or the shot is too tight to include it

The score is present over present, absent and contradicted; occluded is left out of the denominator, since a frame that looks away from where a thing would be is not evidence that it is missing.

Pairwise identity asks whether a named character still looks like the same person several clips later, and its unit is the pair. One call receives the brief's description of the character (on ReelBench, the description in the system's own cast, where it has one), a script excerpt around the two shots and the two frames; it first decides whether the character is visible in each, then rates agreement from 1 to 5 on face, hair, wardrobe and build, on a scale anchored so that a scripted change of appearance is not drift. Ratings are rescaled to $[0, 1]$ and weighted equally per character and per pair within a character. A pair in which the character is absent from either frame is dropped rather than scored

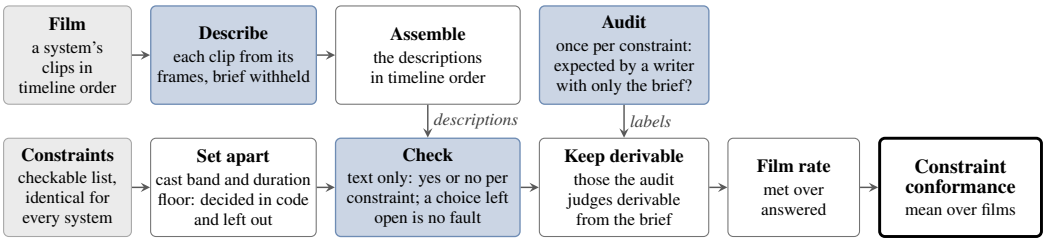

Figure 10: ReelBench video conformance on the 404 premise-derivable story-specific constraints. A text-only checker evaluates blind clip descriptions against these constraints; the fraction met among answered constraints is averaged over films. All systems receive the complete constraint list, while this diagnostic excludes the remaining story-specific constraints and the code-decided cast-band and duration requirements. Shading as in Figure 7.

as a mismatch, so a system whose characters are less often recognisable is scored on fewer pairs. On identical ViStoryBench inputs identity differs by 0.004. On the consistency panel's scene axis, byte-identical films scored in two runs differ by 0.017, a difference indistinguishable from zero (12 films), and asking the judge again about EvoReel's films changes a film's value by 0.004 on average (41 films). The scene axis is near its ceiling for EvoReel, whose render prompts carry the script's setting text, so part of its value reflects prompt fidelity.

Constraint conformance checks a ReelBench film against the task's checkable constraints. The cast band and duration floor are evaluated in code, with duration measured from the rendered film. For story-specific constraints, each clip is described blind, and a text-only checker answers one closed question per constraint against the assembled descriptions.

The reported video-conformance diagnostic uses the 404 story-specific constraints classified as derivable from the premise (Appendix B). For each film, it computes the fraction met among answered constraints and then averages over films. All systems receive the complete constraint list; the derivability filter applies only to this diagnostic. It excludes the other 115 story-specific constraints and the code-decided cast-band and duration requirements, and therefore does not measure satisfaction of the full task specification.

The blind panel of Figure 4(c) compares an adopted repair with the clip it replaced. Raters see the script line and each clip's frames as an eight-by-four grid, sides assigned by a hash of the two clip identifiers, and are not shown which side is the repair, the commitment rates or the repair instruction. The three raters of each ReelBench shot are prompted as a clause-by-clause checker, a visual-effects supervisor and an ordinary viewer and share no context; a ViStoryBench shot has one to three ratings. A vote for the repair counts 1 and a tie 0.5, averaged within a shot and then over shots, and each rate is compared with 0.5 by a one-sample t-test over shot means, which on ReelBench, where every shot has three ratings, agrees with the test specified for the panel before it was run, a cluster bootstrap over shots. Figure 11 probes, with a single rater, how far a verdict depends on the frames shown.

C.5 Further video columns and coverage

Table 5 reports the video columns that Table 2 does not carry: under the proprietary renderer, place consistency (Section 4.2), execution and strict completion on ViStoryBench; under the open-weight renderer, execution, completion, strict completion and identity on ViStoryBench, execution on MovieBench and conformance on ReelBench. EvoReel's films are rendered from the scripts scored in Table 2, and its open-weight films in Table 5 do not pass through the video self-judge.

Because every rate except completion is taken over the shots a system delivered (Section 4.1), coverage differs by system. On ViStoryBench under the open-weight renderer, over the 76 stories every system covers, execution rests on 1,252 shots for EvoReel and on 1,170 to 1,207 for other baselines. The shots a baseline lacks are neither easier nor harder for EvoReel than the rest: restricted to any one baseline's shots, EvoReel's rate lies between 0.705 and 0.713. On MovieBench, EvoReel and Anim-Director are scored on the same 37 of the 41 films, and execution rests on 2,587 shots for EvoReel and on 1,873 to 2,545 for Anim-Director, VideoDirectorGPT and ViMax; on the

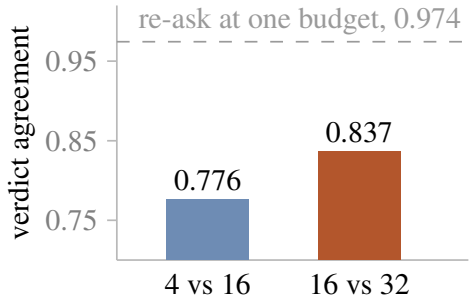

Figure 11: How far a pairwise video verdict depends on the number of frames shown, over the same 196 pairs with clips, blinding and rater held fixed. Agreement is 0.776 between 4 and 16 frames and 0.837 between 16 and 32, the panel's budget, both below one budget's agreement with itself (0.974, dashed); the probe used a single rater rather than the three-agent panel of Figure 4(c).

Table 5: Video columns not in Table 2. *Cons.* is place consistency (nulls in Section C.4, provenance control in Section 4.2); *Exec.* the share of a shot's annotated characters, actions and props that its clips show, taken for EvoReel under Veo with the video self-judge off, since the self-judge selects clips on this rate; *Compl.* and *Strict* the share of the brief's shots shown, with and without half credit for a partly shown shot; and *Conf.* the conformance metric of Table 2's ReelBench Shown column, here on the LTX-2.3 films with EvoReel's video self-judge off. Completion is judged from more frames per clip under Veo than under LTX-2.3 and is not compared across renderers. Under LTX-2.3, ViStoryBench columns cover the 76 films every scored system shares (38 for identity), MovieBench 37 and ReelBench 104 (84 for Mora). Shading marks the best, second and third system in each column.

	ViStoryBench Veo 3.1 Fast			ViStoryBench LTX-2.3				MovieBench LTX-2.3	ReelBench LTX-2.3
Method	Cons.	Exec.	Strict	Exec.	Compl.	Strict	Ident.	Exec.	Conf.
ScriptAgent	0.692	0.594	0.106	0.571	0.198	0.001	0.716	0.612	0.349
VideoDirectorGPT	0.786	0.660	0.169	0.623	0.527	0.159	0.816	0.680	0.387
Anim-Director	0.774	0.678	0.225	0.688	0.608	0.271	0.774	0.701	0.409
VideoAgent	0.725	0.618	0.116	0.605	0.445	0.029	0.811	0.642	0.417
ViMax	0.731	0.645	0.175	0.634	0.562	0.222	0.804	0.665	0.378
MovieAgent	0.617	0.534	0.053	0.553	0.421	0.066	0.712	0.589	0.328
Mora	0.648	0.542	0.211	0.582	0.506	0.196	0.705	0.605	0.303
Co-Director	0.851	0.686	0.259	0.680	0.624	0.295	0.729	0.658	0.410
EvoReel	0.974	0.747	0.347	0.710	0.663	0.353	0.872	0.845	0.426

1,931 shots that both EvoReel and Anim-Director score, EvoReel's rate is 20.9% higher than Anim-Director's. MovieBench identity scores a film only where a named character is found in two of its clips, so Table 2 averages it over the 12 films every system scores, where Anim-Director leads and EvoReel is second.

On ReelBench under the proprietary renderer, conformance covers 98 to 104 films per system (Anim-Director rendered 99), since a film drops out when none of its frame-decided constraints receives an answer. Conformance has a target shared by every system, the benchmark's constraints, but identity does not, because a premise names no cast; it is the one column scored against each system's own plan, which is why Table 2 does not rank systems on it. It is scored on 100 of EvoReel's films and on 38 to 85 of each baseline's, because a character enters only when it is found in two of a film's 12 screened clips. Co-Director, Mora and VideoAgent record no cast, so a language model reads one off each film's own shot prompts; on EvoReel's films, whose recorded cast is known, such an extracted cast moves identity by 0.005. Their lead characters are recognised in about 20% of screened clips against 73% for EvoReel, so their 43 to 48 scored films are those in which the lead character was recognised, and a pair drawn on a misrecognised person counts as an inconsistency.

MovieAgent's plans name its characters without describing them, so its identity is scored against the names alone; a cast read off its prompts does not move it detectably on the films both score. The identity null on identical media was measured on ViStoryBench (Section C.4), not on ReelBench or MovieBench.

EvoReel's films keep the brief's shot count, whereas every baseline except ScriptAgent plans a median of 1.4 to 2.0 times as many shots as the brief lists. No video score is adjusted for film length, and places and characters may be easier to keep consistent over fewer shots.

Completion counts every shot of the brief, so a film with no clips scores zero. On ViStoryBench under the proprietary renderer this happens in 11 of the 720 cells, seven of them ScriptAgent's and one EvoReel's; with those cells dropped, EvoReel still leads every baseline, by at least 0.077, and by at least 0.092 on the share of shots fully met. Each arm's completion is scored in one invocation of its own, never joined from runs made at different times, so EvoReel's films and the baselines' were judged there in two invocations a day apart, with the same model, question bank and sampling, each run once. In a separate re-judging test on 12 of these stories under the same renderer, at 2 frames per clip, byte-identical media changed 12.7% of shot statuses, and a difference of 0.019 between two EvoReel configurations over those films was within this noise. Completion is judged from 12 frames per clip under the proprietary renderer and from 2 under the open-weight one, so its values under the two renderers are not comparable.

D EvoReel in detail

EvoReel's two self-judges are separate components with separate evidence: the script self-judge scores a draft and drives the writing loop, and the video self-judge checks a rendered shot against its script line and decides whether it is retried. The prompt excerpts below are quoted as run.

D.1 Script-side loop

The script self-judge of Section 3.1 scores seven dimensions on $[0, 5]$, D1 to D7 in the order of Table 6, with one prompt for open briefs and another for shot-list briefs. A dimension scoring below $\tau = 3.5$ is flagged, and the loop runs at most three rounds. A fixed map, not the model, assigns each dimension to the writing role responsible for it, D1, D2, D3 and D7 to the director, D4 and D5 to the screenwriter and D6 to the shot writer, and Section 4.4 counts flags by this map. Experience notes enter only the writers' prompts, so the judge's prompt is the same in every arm. Four dimensions carry a hard cap, so that a failure the loop exists to catch cannot be averaged away by a favourable overall impression:

> *D3 Structure & Pacing.* CRITICAL CHECK: scan the middle — are there $\geq$2 consecutive scenes that add NO new information / obstacle / choice (a collapse)? If so this dimension cannot score above 2.5. Cite the offending shots.
> *D4 Character.* CRITICAL CHECK: if characters are only costumes executing plot mechanics with no inner life, score $\leq$ 2.5.
> *D5 Theme & Coherence.* If the payoff is asserted-not-earned, score $\leq$ 2.5 and say which shots plant vs pay off.
> *D7 Brief Constraint Satisfaction.* CRITICAL CHECK: if required characters or specified beats are MISSING, score $\leq$ 2.5 and NAME exactly what is missing.

Figure 5(a) and Table 6 combine two runs of the script self-judge on ViStoryBench, both without experience. The flag rates are taken over the 188 rounds, on 79 of the 80 stories, in which the judge returned all seven scores. The other 53 logged rounds carry no verdict and are excluded: in all but one of them the edit step committed nothing, so the judge was not called again. The paired change comes from the second run, which runs the loop alone from the same first drafts on 78 stories, and compares each story's draft verdict with its final one, so a story the loop left unchanged contributes a zero instead of dropping out. In that run the first round commits an edit on 50 stories and writes a visual treatment on 268 shots, all into empty fields, at a median of 365 characters. The second and third rounds commit on 34 and 30 stories and mostly rewrite existing treatments, so the median net growth of a treatment they change falls to 69 and then 19 characters.

Table 6: The script self-judge on ViStoryBench: share of the 188 scored rounds in which each dimension scores below $\tau = 3.5$, and mean paired change in its score (0 to 5) from draft to final verdict over 78 stories, with a paired Wilcoxon test. Rows are D1 to D7 in order.

Dimension	Flagged (of 188)	Paired change (n=78)	p
Story and hook	94%	+0.042	0.008
Conflict and stakes	90%	+0.015	0.359
Structure and pacing	93%	+0.054	0.052
Character	100%	+0.062	0.002
Theme and coherence	97%	+0.067	0.002
Shot craft	**41%**	**+0.162**	$< 10^{-4}$
Brief coverage	0%	+0.036	0.020

D.2 Video self-judge

The video self-judge of Section 3.2 applies the construction of execution (Section C.4) to a single shot against EvoReel's own script line rather than the benchmark's text. The line is decomposed into at most eight commitments, each tagged as story-bearing or decoration, the clip is described blind, and one text-only call answers every commitment, with no replicates, so the check is affordable at every shot. Because the blind describer never names characters, the checker is told to match a named character to the figure whose appearance fits, and that a paraphrase counts, without which the check collapses into lexical overlap:

> A PARAPHRASE COUNTS. "lifts the baby rabbit up in the air" satisfies "picks up Little Brown Rabbit"; "a window showing a full moon" satisfies "the moon is visible through the window". Answer "no" only when the description contradicts the commitment, or when it says nothing that bears on it at all. Wording never has to agree; the observable fact does.

In this form, with Gemini 3.5 Flash describing and Claude Sonnet 4.6 decomposing and checking, the check agrees with GPT-5.5 as an alternative checker at $\kappa = 0.674$ over 3,157 commitments on 436 ViStoryBench shots. A shot whose check returns no parsable answer keeps its first render and is not repaired. A shot is retried when its commitment rate in Equation 3 falls below $\theta = 0.60$, worst first and at most $\lceil \rho n \rceil$ of a film's n shots ($\rho = 0.25$). On ReelBench most first renders fall below the threshold, so the cap rather than the threshold sets how many shots are retried. Figure 5(b) compares the two renders of every retried shot whose check returns a usable record for both, over all such shots and over those whose second render is adopted; since the clauses come from the shot's script line, they are identical on both sides.

D.3 Experience memory

Of the 4,846 notes in the experience memory, 4,529 are specificity notes and 317 are omission notes. Among the specificity notes, 3,540 fill an empty visual treatment and carry no verdict (Section 3.3); retrieval serves them in the same block, under the same header, as the revisions that raised specificity. The rest come from rewrites of an existing treatment. Each rewrite is labelled by the change in that shot's specificity score, positive when it is at least 0.3 and negative when it is zero or below; a rewrite whose change falls in between is not stored, and the stored rewrites comprise 301 positive and 688 negative notes. Each scene requests five notes from $\mathcal{M}^{+}_{\text{spec}}$ and five from $\mathcal{M}_{\text{omit}}$ (Equation 4) together with the closest negative note, and the length budget then trims the request, retaining the negative note in 1 of the 5,208 requests that retrieved one. The retrieval comparison of Section 4.4 runs over the 104 ReelBench tasks, and both of its arms ask for the visual treatment at construction. Retrieval adds 0.144 camera-movement terms per shot, more on 94 tasks and fewer on 7. Copying is measured as 8-gram overlap between the treatments and the retrieved exemplars, against a floor of 0.003% obtained by scoring the same exemplars against the treatments of the arm that never retrieved them.

E FURTHER FRAME AND SCRIPT COMPARISONS

Figure 12 to Figure 14 add six ViStoryBench films under the proprietary renderer, drawn as in Figure 6 but with EvoReel and all five baselines whose clip-to-shot mappings are validated. Each film comes from a rule fixed before any frame was viewed, and the rules fall into four groups, from rules that favour EvoReel through a median rule and a seeded random draw to one that favours a baseline. Three films come from rules that favour EvoReel: the rule of Figure 6(a), which ranks films by EvoReel's own completion and place consistency, returned Stories 36, 20 and 3 besides Story 10, and EvoReel's completion margin over Co-Director, among the 70 films both cover in five or more shots, is largest on Stories 52, 3 and 54. Stories 20, 3 and 54 were shown after all five films these two rules returned besides Story 10 had been viewed. Story 37 is one of six films that a median rule returned, the three whose margin over Co-Director lies closest to its median over all 80 films and three of the four whose margin over Anim-Director equals its median, and was shown after all six were viewed. Story 18 is the first of three films that a generator with a fixed seed drew from the 67 in which EvoReel, Co-Director, Anim-Director and VideoDirectorGPT each have clips for five or more of the brief's shots, shown without redrawing. Story 6 is the film on which Co-Director's completion most exceeds EvoReel's among the 70 films of the margin rule.

Except on Story 6, whose window is the whole film, each rule fixed five shots on EvoReel, Co-Director, Anim-Director and VideoDirectorGPT, preferring five consecutive shots that all four cover; where none exist, the window holds the most covered cells and its caption calls it a fallback. ViMax and MovieAgent play no part in choosing the windows. A tile shows the first clip the judge cited for its shot, even where the clip-to-shot mapping assigns that clip to another shot; the mapping decides a tile only where the judge cited none, and it never enters a status. It places EvoReel's clips through its render record and most baseline clips by text alignment, MovieAgent's on every film. The judge cited multiple clips for 30 of the 150 baseline cells shown and none of EvoReel's 30. For a shot carried forward as *met*, the third-stage check uses sampled frames from all cited clips, whereas the tile shows only the first cited clip's middle frame. Burned-in subtitles appear in two of these tiles (Figure 13(b) and Figure 14(b)) and can count in a clip's favour, because the judge reads them as dialogue (Figure 14(b)); EvoReel's shot-6 clip in Figure 6(a) also carries one, outside the frame shown.

These films illustrate what completion counts, not how often one system does better; the per-film counts over all 80 films are those of Section 4.5. Among the main-text films, Story 27 of Figure 6(b) was chosen after viewing the six films of its rule, partly for a status difference visible in its tiles.

Because ReelBench's conformance and identity columns are not keyed to brief shots, any system can be shown there: Figure 15 follows each shown system's own lead character on two premises, and Figure 16 samples each shown system's whole film on two others. In both, the candidate rows were fixed before the premises were picked, and each caption names the baselines left out. Neither figure ranks systems.

E.1 SCRIPT EXAMPLES

Table 7 and Table 8 set side by side the opening entries of the scripts that EvoReel and two baselines write for one ViStoryBench story and one ReelBench premise, to show what the two script axes reward; on ViStoryBench all three entries are written for the brief's first shot. Production specificity, judged without the brief, rewards entries that fix how each shot is framed, lit, staged and timed; content fidelity rewards keeping what the brief asks for, beat by beat on ViStoryBench, where its judge is told to leave the camera setup aside, and constraint by constraint on ReelBench. Entries are quoted from the text the script judges scored, which writes each plan as numbered descriptions with tags for camera, the characters in the shot, duration and any dialogue, together with the fields each system's plan defines, for EvoReel a shot size and a visual treatment. Each whole-script score is the mean of five judge samples, eight for the baselines' ViStoryBench fidelity, and EvoReel's are those of the scripts scored in Table 4 (Section D.1).

Each example is the task on which EvoReel's margin on one axis over two named baselines is largest, chosen to illustrate that axis rather than as a typical case. On ViStoryBench the two are Co-Director and Anim-Director, the baselines of Figure 6, chosen there as the two with the highest completion, whose scripts rank last and third among the baselines on the ViStoryBench composite;



Figure 12: ViStoryBench films chosen by rules that favour EvoReel (Appendix E), drawn as in Figure 6 with EvoReel and the five baselines whose clip-to-shot mappings are validated. **(a)** Story 54, shots 1 to 5, third by EvoReel's completion margin over Co-Director; a fallback window. ViMax ties EvoReel's summed credit over these shots (4.0). **(b)** Story 20, shots 4 to 8, second of the four films returned by the rule of Figure 6(a). EvoReel's credit is the highest (4.0, against Co-Director's 3.0). MovieAgent's shots 4 and 5 show one clip, judged missed and partial. The briefs retell Hemingway's *The Old Man and the Sea* (a) and the television drama *Soldiers Sortie* (b).



Figure 13: Two more films, drawn as in Figure 12. **(a)** Story 3, shots 7 to 11, third of the four films of the rule of Figure 6(a) and second by margin over Co-Director; a fallback window. EvoReel has the highest credit (4.0, against Anim-Director's 3.5), and its Zhen Huan wears shot 7's light green and, in shots 8 and 11, her cast entry's lavender. VideoDirectorGPT's shot-9 frame is a zoom blur. **(b)** Story 37, shots 1 to 5, one of the films whose margin over Anim-Director equals the median (Appendix E); a fallback window. ViMax ties EvoReel's credit (3.5). The grey cells mark a missed shot: Anim-Director's and ViMax's films contain no clip for shot 4, so both miss it. VideoDirectorGPT renders shot 4 but also misses it, and its frame carries a burned-in subtitle. The briefs retell *Legend of Zhen Huan* (a) and Hao Jingfang's *Folding Beijing* (b).

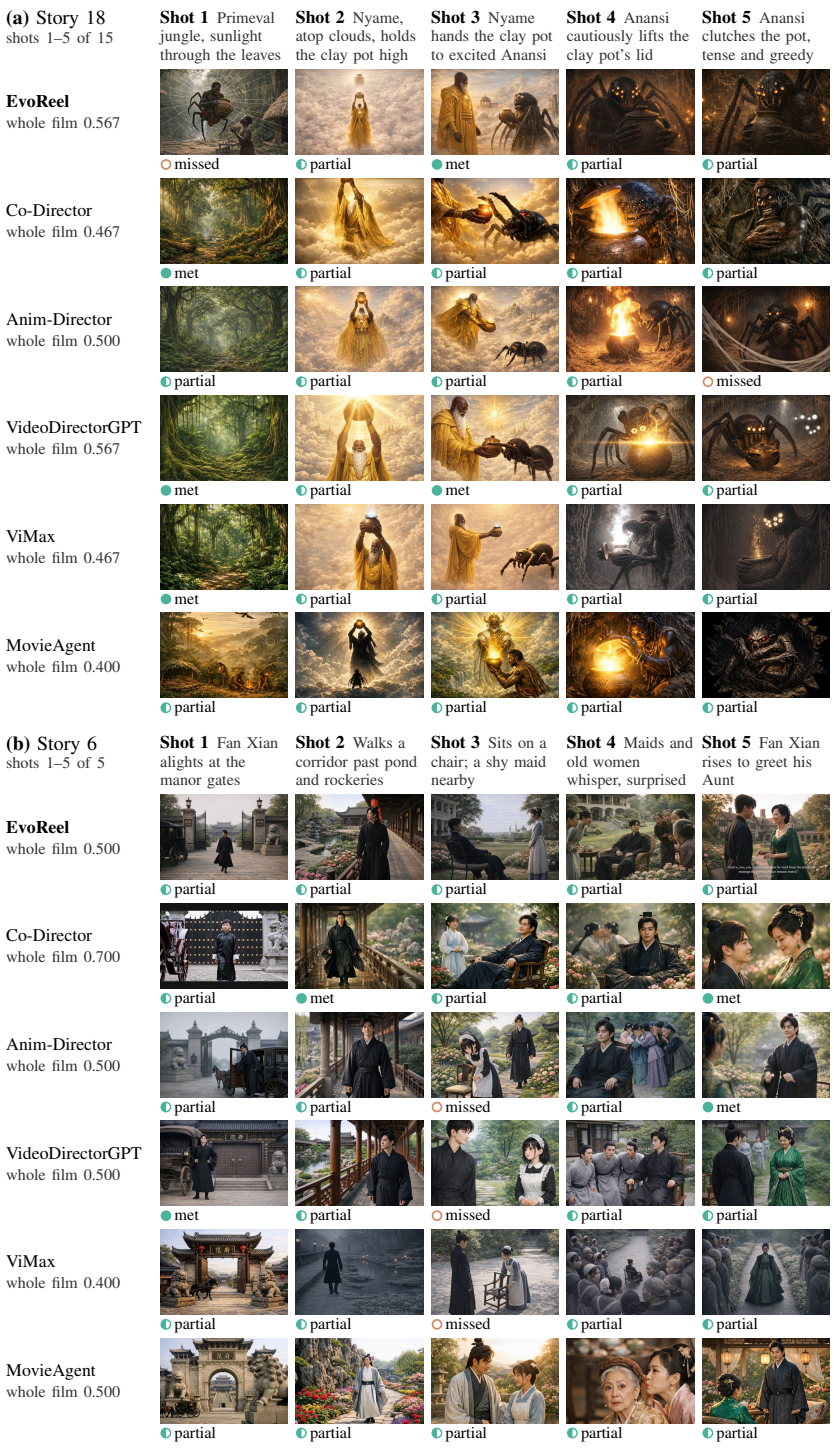

Figure 14: Two more films, drawn as in Figure 12. **(a)** Story 18, shots 1 to 5, the first film of the seeded random draw; a fallback window. Over these shots VideoDirectorGPT's credit is 3.5, Co-Director's and ViMax's 3.0 and EvoReel's 2.5. EvoReel's shot-1 tile is its first mapped clip, since the judge cited none. **(b)** Story 6, the whole film, on which Co-Director's completion most exceeds EvoReel's. Each of EvoReel's five shots is judged partial, and Co-Director meets shots 2 and 5. All three replicates read the burned-in subtitle of EvoReel's shot 5 as dialogue. The brief retells *Joy of Life*.

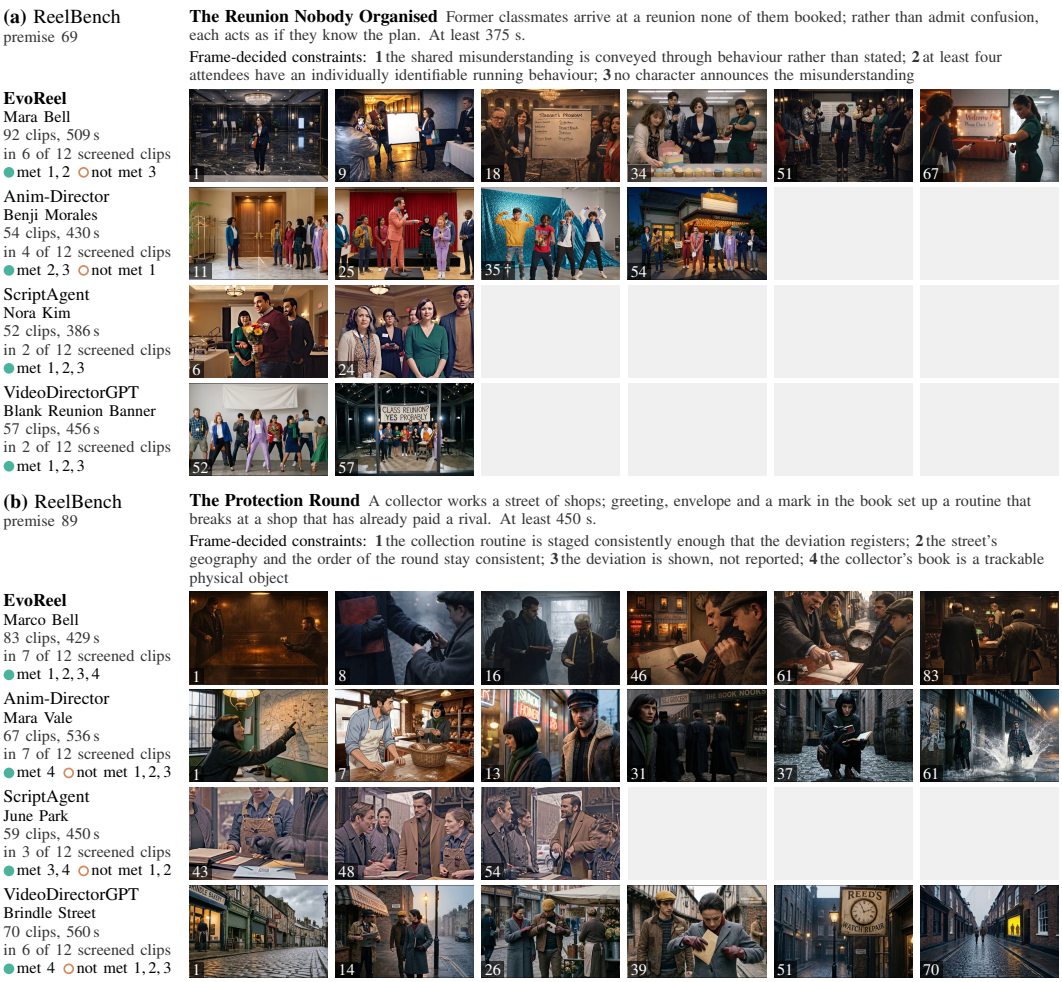

Figure 15: Each system's own lead character on two ReelBench premises under the proprietary renderer, each panel headed by a shortened brief. A premise names no cast, so each row follows one of the up to three characters the identity column scores for that system, the one with the most scored pairs. Its tiles are the middle frames of up to six of the clips, among the 12 screened per film, in which the identity screening found it, spread evenly in clip order, and the number in a tile is the clip's position in the film; a grey cell means it was found in fewer, and a dagger marks a clip in which the pairwise-identity judge found it absent from every pair. VideoDirectorGPT's plans list props and places as cast, so its entries are a banner (a) and a street (b). The header lists the premise's constraints that are decided from frames and derivable from the brief, and each row label gives the film's clip count and length and that system's verdict on each constraint, which covers its whole film rather than the clips shown. Identity is scored here against each system's own cast, and the figure orders no systems. Premises 69 (a) and 89 (b) are the third and second of four returned by a rule on EvoReel's own identity and conformance ranks, over the premises where at least three of Anim-Director, ViMax, ScriptAgent and VideoDirectorGPT have an identity score, and were shown after viewing all four; premises 87 and 61 are not shown, and the baseline rows are the first three of those four with a score. Co-Director, Mora and VideoAgent, whose casts are read off their prompts, and MovieAgent, whose plans name characters without describing them (Section C.5), are not among the candidates. The rule favours premises on which EvoReel's verdicts are good: on premise 89 it meets all four, as on 19 of its 101 films, and its film runs 429 s against the premise's 450 s floor, which conformance does not score.



Figure 16: Each system's film on two ReelBench premises under the proprietary renderer, each panel headed by a shortened brief. A tile is the middle frame of one of six clips spaced evenly from the film's first to its last and numbered by its position in the film; row labels give each film's clip count, length and verdicts on the header's frame-decided constraints. A verdict covers the whole film and rests on blind clip descriptions that are not stored, so the tiles do not explain it: ViMax's and Co-Director's tiles in (b) show long queues, yet both films fail constraint 1. The rows were fixed before the premises; MovieAgent, Mora and VideoAgent are not shown, and Mora, too, meets all five on premise 91. The premises were fixed before any frame was viewed: of the 19 on which EvoReel meets every such constraint, the two with the lowest mean conformance of the five baselines shown, leaving out Figure 15's (premise 89 ties 52). The rule favours EvoReel, whose overall conformance is level with Anim-Director's and ScriptAgent's (Table 2). EvoReel's film in (a) runs under the premise's 450 s floor, which conformance does not score; ScriptAgent's first frame in (b) carries a burned-in subtitle.

Table 7: The first shot of ViStoryBench Story 39 and the entry each of three scripts writes for it, its own first shot, quoted verbatim, one field to a line. The first box is the shot as every writer receives it, with no character annotated. EvoReel's entry opens with the brief's setting line, copied word for word, and restates every fact of the narration in a description that places Weilong Ge in the frame; Co-Director's keeps part of the setting and, of the narration, says only that the house is Mrs. Liang's and that Weilong is about to enter this world of wealth; Anim-Director's keeps every visual element of the setting but, of the narration, only Mrs. Liang's name. Anim-Director's camera tag holds the motion field of its plan; Co-Director's plan has no camera field, and the keyframe prompts it writes for its image model are not part of its script. Over the whole script, content fidelity is 4.80 for EvoReel and 1.00 for both baselines, and production specificity is 3.78, 2.50 and 3.62 for EvoReel, Co-Director and Anim-Director. Among the 59 films whose every brief shot has a clip in EvoReel's rendered film, Story 39 is the one on which EvoReel's fidelity exceeds the better of these two baselines' by the most, 3.80. The brief retells Eileen Chang's *Aloeswood Incense: The First Brazier*.

ViStoryBench Story 39, brief shot 1 of 23 (every writer's input; no character annotated)
1. [Daytime, a luxurious residential area in Hong Kong, modern-designed houses with green-glazed tile roofs, green glass windows framed in chicken-fat yellow, tidy streets, bright sunlight] Weilong Ge, an ordinary Shanghai girl, has lived in Hong Kong for two years. One day, she visits her wealthy aunt, Mrs. Liang, for the first time. A panoramic view of the luxurious residential area, a modern-designed house at the center of the frame, the green-glazed tile roofs sparkling in the sunlight Camera: Wide shot, eye level shot.

EvoReel (shot 1 of 23)
1. wide
Daytime, a luxurious residential area in Hong Kong, modern-designed houses with green-glazed tile roofs, green glass windows framed in chicken-fat yellow, tidy streets, bright sunlight. Daytime in a luxurious residential area in Hong Kong: tidy streets sit under bright sunlight, lined with modern-designed houses whose green-glazed tile roofs glitter sharply. At the center of the eye-level panoramic frame stands Mrs. Liang's modern house, its green glass windows set in chicken-fat yellow frames; Weilong Ge, an ordinary Shanghai girl in her light blue Nanying Middle School top and dark blue skirt, appears small within the broad view as she visits her wealthy aunt for the first time after two years living in Hong Kong.
Visual treatment: Use an eye-level wide composition with the modern house anchored precisely at center, letting the sparkling green-glazed roof and green glass windows draw the eye. Bright direct sunlight keeps the street crisp and orderly, while Weilong remains a modest figure within the larger residential panorama.
camera: static, eye-level | who: Weilong Ge | ∼8s

Co-Director (shot 1 of 23)
1. Establish a sunlit luxury residential district in Hong Kong, revealing Mrs. Liang's modern house with green-glazed tile roofs at the center, setting the world of wealth Weilong is about to enter.
∼7s

Anim-Director (shot 1 of 23)
1. Wide eye-level panoramic view of a luxurious Hong Kong residential area in bright daylight. Modern houses with green-glazed tile roofs and green glass windows framed in chicken-fat yellow line the tidy street, with Mrs. Liang's elegant house centered in frame and the roof tiles sparkling in the sun.
camera: Slow establishing pan across the upscale neighborhood, settling on Mrs. Liang's house. | ∼6s

the rule runs over the 59 films whose every brief shot has a clip in EvoReel's rendered film. On Story 39, the film the rule returns, EvoReel's fidelity is the highest of all nine systems, followed by VideoDirectorGPT's 2.75, and ScriptAgent's specificity is the highest, 4.28 against EvoReel's 3.78. Premise 17 is the one on which EvoReel's specificity most exceeds that of Anim-Director and ScriptAgent, the two baselines with the highest ReelBench script composite. There EvoReel's 4.70 is the highest of all nine systems, followed by VideoDirectorGPT's 4.44; on fidelity ScriptAgent's script reaches the maximum of 5.00, and Co-Director's, VideoAgent's and VideoDirectorGPT's are also above EvoReel's 4.40.

Table 8: Opening shots of three scripts written for ReelBench premise 17, quoted verbatim, one field to a line. The first box shows the premise; every system also receives its complete list of eight checkable constraints, omitted here. Content fidelity evaluates this same list, which EvoReel's script self-judge also uses to assess each draft (Section 3.1). Anim-Director's camera tag holds the motion field of its plan, and EvoReel's scripts on this corpus have no dialogue field. EvoReel's entry fixes shot size, lens, framing, light and camera, whereas the baselines' entries give setting and action, and ScriptAgent's a line of dialogue. Over the whole script, production specificity is 4.70, 3.26 and 3.42 for EvoReel, Anim-Director and ScriptAgent, and content fidelity, conformance to the eight constraints, is 4.40, 3.00 and 5.00. The three scripts have 57, 36 and 34 shots; script length differs by system on ReelBench (Section 4.2). All three systems finished a script for each of the 104 premises, and premise 17 is the one on which EvoReel's specificity exceeds the better of these two baselines' by the most, 1.28.

ReelBench premise 17: brief (every writer's input)
A comedy two-hander. An engaged pair try repeatedly to announce that the wedding is cancelled, and every attempt is derailed before it lands. Their rehearsals get shorter and more desperate as the evening proceeds. The comedy is structural, so the pattern of attempt and derailment must be clear and must escalate rather than repeat. Carry between 3 and 4 named recurring characters; at least 225s.

EvoReel (shot 1 of 57)
1. wide
In a narrow cream restaurant hallway, June Park and Max Adler stand close together outside closed double doors. June wears a neat black dinner dress with a pearl hair clip; Max wears a rumpled navy suit with a loosened tie. Warm sconces line the wall, and party light glows from beneath the doors while they huddle over a stack of cue cards.
Visual treatment: Locked-off 35mm hallway framing keeps the double doors centered deep in the composition, with June Park and Max Adler compressed between cream walls. Warm sconce light and the glow under the doors make the corridor feel like a backstage threshold.
camera: static | who: June Park, Max Adler | ∼4s

Anim-Director (shot 1 of 36)
1. Nora and Eli stand at the entrance of the Grand Ballroom, smiling stiffly at the festive crowd while a huge banner behind them reads 'Tomorrow: Nora & Eli Forever.' Their eyes say panic, not romance.
camera: Aunt Marge waves them in with her clipboard; Dev tests a microphone with a soft feedback squeal as Nora and Eli exchange a doomed look. | who: Nora, Eli, Aunt Marge, Dev | ∼8s

ScriptAgent (shot 1 of 34)
1. In the catering kitchen behind the hall, Nora stands beside a tray of untouched mini quiches, smoothing a folded index card. Eli blocks the swinging door with his back and checks the tiny window for guests.
∼10s
dialogue:
Nora: "We say it once, clearly, before the toast: the wedding is canceled, but we are okay."