BVB: BENCHMARKING AGENTIC VIDEO UNDERSTANDING VIA PROGRAMMATIC RECONSTRUCTION IN BLENDER

Anonymous authors
Paper under double-blind review

ABSTRACT

Multimodal agents can create complex videos in software such as Blender by writing code instead of using diffusion models. Yet video understanding benchmarks still evaluate models mainly through question answering. If an agent truly understands a video, it can reconstruct it programmatically. We introduce **BVB**, Blender-VideoBench, a benchmark that tests this ability by asking agents to reconstruct real-world videos as animated Blender scenes. To ensure fair comparison, each agent programs the reconstruction through a lightweight harness, **Mini-BVB**, in an identical sandbox under a shared cost limit. The benchmark renders each reconstruction from its animated camera and evaluates it on two axes: (1) **Dual VQA** measures how many spatiotemporal facts the reconstruction preserves. (2) **Latent Similarity** measures how closely the reconstruction matches the source video perceptually. Our overall score, a square-root mean, favors balanced performance. We evaluate 51 configurations from 10 model families and analyze semantic retention, perceptual similarity, reasoning effort, and cost. The best model reaches 88.6 Latent Similarity but retains only 53.7% of the spatiotemporal facts from the source video. Additional reasoning improves perceptual similarity but does not close this gap. In a blind study with 15 raters and five configurations, Latent Similarity correlates strongly with human preference. These results show that programmatic reconstruction is a viable test of agentic video understanding, and that semantic retention remains the main challenge.

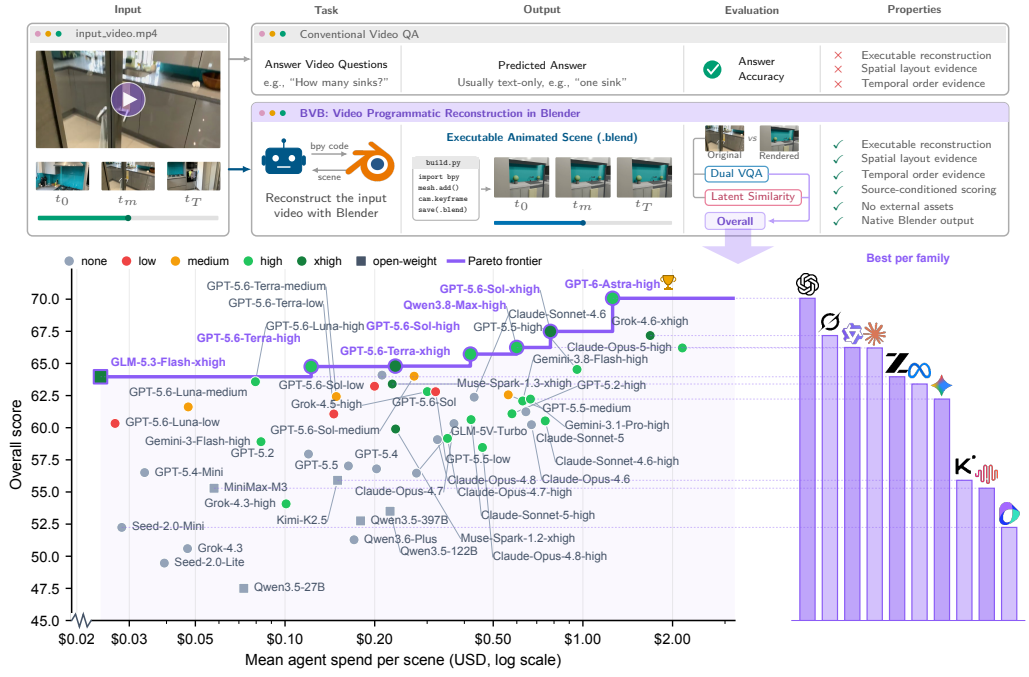


Figure 1: BVB overview and **Overall** cost frontier across 51 configurations.

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.

1 INTRODUCTION

Video understanding is usually measured by question answering (Tang et al., 2025a; Jang et al., 2017; Lei et al., 2018; Yu et al., 2019; Xiao et al., 2021; Pătrăucean et al., 2023; Li et al., 2024; Wu et al., 2024; Fu et al., 2025; Tang et al., 2025b), but a correct answer can come from answer priors or a single frame (Lei et al., 2023), so it does not show that the model tracked the scene over time. *If an agent truly understands a video, it can reconstruct it programmatically.* A reconstruction must approximate the source in object placement, camera trajectory, and the order in which objects appear. These details cannot be guessed from a single frame or answer prior. The agent needs to work from video frames alone, without depth maps, segmentation, or 3D ground truth, so it must infer the full scene from pure 2D observation. To rebuild a video, an agent must combine spatial, temporal, and compositional understanding with reasoning and coding. Existing video benchmarks test these abilities separately.

This test has recently become possible. Multimodal agents can now construct visual content by writing code instead of using diffusion models. Recent systems build animated Blender scenes through agent-driven code (OpenAI, 2026; Ricouard, 2026; Yin et al., 2026; He et al., 2025; Ahuja, 2025), suggesting that these agents already have some critical spatiotemporal understanding capabilities. However, existing results come from selected scenes, often with repeated human guidance and external asset libraries, so they do not show how reliably an agent can handle new scenes, how performance changes across model families, or how well the resulting scene matches the source in layout and dynamics. A rigorous benchmark should evaluate this ability at scale through holistic reconstruction of real-world videos under a shared protocol without external assets.

To enable such controlled evaluation, we introduce **BVB**, Blender-VideoBench, a benchmark asking multimodal agents to reconstruct real-world videos as animated Blender scenes, as shown in Figure 1. To ensure that agents are required to understand actual scenes, and to keep reconstruction complexity manageable while preserving rich spatiotemporal structure, we construct the benchmark with the egocentric real-world indoor videos and question-answer pairs from VSI-Bench (Yang et al., 2025). Each agent interacts through a lightweight harness (**Mini-BVB**) that offers two actions, inspecting video frames and executing code in a Blender sandbox, under a shared cost limit. External asset libraries are disallowed, so the agent must construct the scene from primitives, animate its camera along the source trajectory, and save the result as an editable Blender file instead of a generated image or a pre-rendered video.

A faithful reconstruction must preserve the source video's layout and dynamics, so BVB evaluates each reconstruction along two axes: (1) **Dual VQA** (**DV**) measures how many spatiotemporal facts the reconstruction preserves. It asks a VLM judge the same spatial and temporal questions on the source and reconstruction, from object counts and distances to route plans and appearance order, and scores retention only on questions the judge answers correctly on the source. (2) **Latent Similarity** (**LS**) measures how closely the reconstruction matches the source video perceptually, comparing the two videos with frozen V-JEPA 2.1 representations (Bardes et al., 2024; Assran et al., 2025; Mur-Labadia et al., 2026). We report both axes and rank configurations by their square-root mean (**Overall**), which favors balanced performance across the two axes.

We evaluate 51 configurations across 10 proprietary and open-weight model families and analyze semantic retention, perceptual similarity, reasoning effort, and cost. GPT-6-Astra-`high` leads at 70.07 **Overall** and reaches 88.6 **LS**, but it retains only 53.7% of the spatiotemporal facts from the source video. Current agents therefore build reconstructions that look right but get many facts wrong. Additional reasoning improves perceptual similarity but does not close this gap in semantic retention. To validate the automatic evaluation, we conduct a blind study with 15 human raters, whose mean ranking of five configurations matches the **Overall** order exactly. These results show that programmatic reconstruction is already a viable test of video understanding, but that the best models still miss nearly half the spatiotemporal facts from the source video.

In short, our contributions are threefold:

- We introduce **BVB**, a benchmark that tests agentic video understanding by asking multimodal agents to reconstruct real-world indoor videos as animated Blender scenes under a standardized, cost-controlled, asset-free setup.

- We design a two-axis evaluation that favors balanced performance across semantic retention and perceptual similarity, and validate it against human blind rankings.
- We evaluate 51 configurations across 10 model families, including GPT-6 Astra, and find that current agents produce visually plausible but semantically incomplete reconstructions. These results identify which video understanding abilities remain unsolved, highlighting open challenges for the community.

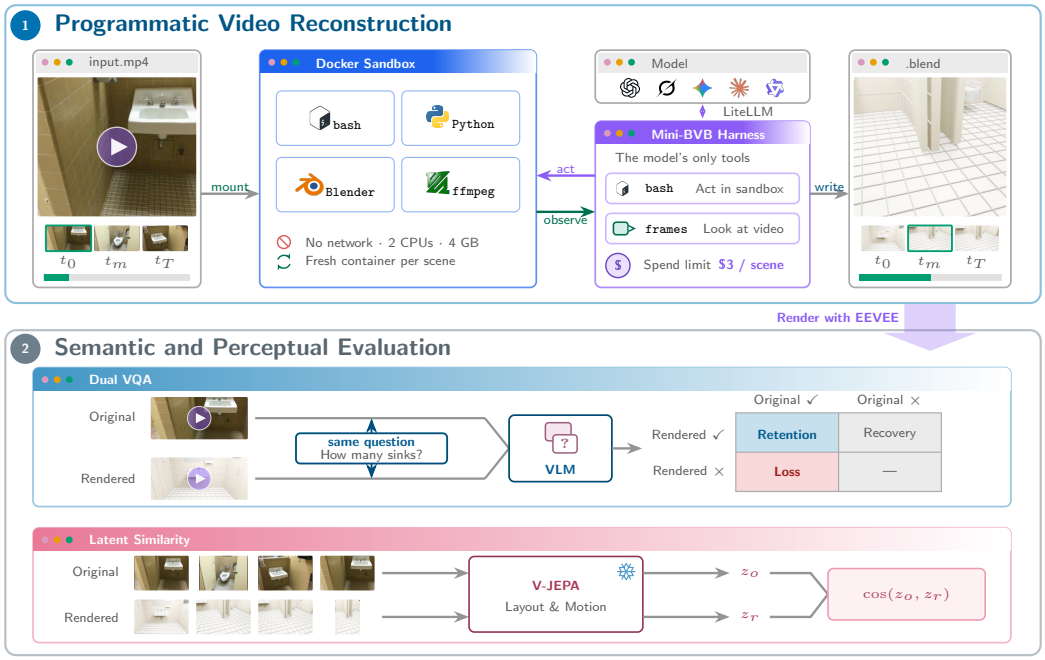


Figure 2: **BVB agent runs and evaluation.** A model alternates between viewing source frames and running Python code in a Docker sandbox under a cost limit, then saves an animated Blender scene. **Dual VQA** measures how well the rendered video retains answers the judge gets right on the source, and **Latent Similarity** compares layout and motion using frozen V-JEPA features.

2 BVB: Blender-VideoBench

2.1 Benchmark Construction

BVB is built on the real indoor egocentric videos of VSI-Bench (Yang et al., 2025), drawn from ARKitScenes (Baruch et al., 2021), ScanNet (Dai et al., 2017), and ScanNet++ (Yeshwanth et al., 2023). We use its 288 scenes and their 5,130 QA pairs, so every reconstruction is scored against the same set of questions. In Stage 1 (Figure 2), each agent reconstructs the video in a Docker sandbox with Blender through a lightweight harness called **Mini-BVB**, which exposes exactly two actions. `bash` runs sandbox commands including Python code for Blender, and `frames` requests video frames by timestamp or uniform count. In this agentic setting, the model itself decides how much of the video to look at and how long to work, with no step or frame budget. The only limit is a per-scene spend cap. External asset libraries are disallowed, so geometry must be built from primitives and basic Blender operations. A run is valid only if the agent observed the source video and produced a renderable Blender file. Stage 2 scores this file and never re-enters the loop. If an agent fails to produce a renderable file, that scene scores zero on every metric, because excluding failed scenes would let a model raise its average by skipping hard ones. Appendix A gives the loop in full and reproduces the system prompt, which is identical for every configuration, so every model faces the same harness, sandbox, prompt, and cost ceiling.

2.2 Evaluation Metrics

BVB evaluates reconstructions on two complementary axes (Figure 2). **Dual VQA** measures how much semantic content the reconstruction retains. **Latent Similarity** measures how closely the reconstruction matches the source in overall visual appearance. We combine both axes into **Overall**.

Dual VQA (DV). A faithful reconstruction should retain the spatial and temporal facts of the source. A VLM judge answers the VSI-Bench questions on uniformly sampled frames of both the source and rendered video. The retention rate of **DV** is defined as $|C_s \cap C_r| \,/\, |C_s|$, where C_s and C_r are the questions answered correctly on the source and reconstruction. We refer to the facts tested by C_s as recoverable facts, since the judge can recover them from the source video. Conditioning on C_s means the metric captures what the reconstruction keeps, not the judge's baseline accuracy.

Latent Similarity (LS). Correct answers to discrete questions do not guarantee that a reconstruction *looks* like the source, so we add a continuous perceptual axis. A frozen V-JEPA encoder (Bardes et al., 2024; Assran et al., 2025; Mur-Labadia et al., 2026) maps both clips to latent features. We compare spatial arrangement and temporal dynamics separately, reporting them as Layout and Motion, and average them to get **LS**. The encoder is a self-supervised video model, never fine-tuned on BVB, so the metric reflects general visual agreement rather than benchmark-specific patterns. Appendix F gives the formal definition and pipeline details.

Overall. The two axes measure different aspects of reconstruction quality. **DV** is a semantic retention rate and **LS** is a perceptual similarity score. Under the arithmetic mean $\bar{s}_i = 1/|\mathcal{A}| \sum_{a\in\mathcal{A}} s_i^a$, a gain on one axis exactly offsets an equal loss on the other. A 10-point increase in **LS** fully compensates for a 10-point decrease in **DV**, which is not the trade-off we want the aggregate to encode. To favor configurations that are strong on both axes we instead use the square-root mean:

$$\bar{s}_i = \left(\frac{1}{|\mathcal{A}|} \sum_{a\in\mathcal{A}} \sqrt{s_i^a} \right)^2, \qquad \mathcal{A} = \{\textbf{DV}, \textbf{LS}\}. \tag{1}$$

Both axes are expressed on a 0–100 scale. Equivalently, the square-root mean is the arithmetic mean corrected by a cross-axis dispersion penalty, $\bar{s}_i = 1/|\mathcal{A}| \sum_{a\in\mathcal{A}} s_i^a - \mathrm{Var}_{a\in\mathcal{A}}(\sqrt{s_i^a})$, as Appendix D derives. Equal gains and losses on the two axes therefore no longer cancel exactly, and more uneven performance receives a larger penalty. Moreover, unlike the geometric mean, the square-root mean does not become zero when only one axis is zero. We therefore use it to rank configurations.

3 Experiments

3.1 Experimental Setup

Models. We evaluate frontier multimodal models as zero-shot reconstruction agents, without fine-tuning. The suite covers ten families of proprietary API and open-weight models, namely OpenAI GPT-5.x and GPT-6 Astra (OpenAI, 2026), xAI Grok, Anthropic Claude, Google Gemini (Gemini Team, 2025), Meta Muse Spark (Meta, 2026), Zhipu GLM (GLM-V Team, 2025; Z.ai, 2026), Alibaba Qwen (Qwen Team, 2025), Moonshot Kimi (Kimi Team, 2025), MiniMax (MiniMax, 2025), and ByteDance Seed (ByteDance Seed, 2025). For models with adjustable reasoning effort, we test the available settings among `none`, `low`, `medium`, `high`, and `xhigh`.

Implementation details. All agents run through the same Mini-BVB harness and Stage 1 sandbox defined in Section 2. Each configuration is evaluated on all 288 scenes using the 5,130 spatiotemporal questions. We use a common $3 per-scene cost ceiling and the same system prompt, which shows the agent its source video and requires an executable Blender program and a final `result.blend`. The Docker sandbox comes pre-installed with Bash, Python, Blender 4.2, and FFmpeg. For **Dual VQA**, the VLM judge is `gpt-5.4-mini`, answering all VSI-Bench questions on 16 uniformly sampled frames from both the source and rendered video. The judge answers 35.6% of the questions (1,827 of 5,130) correctly on the source videos, and these questions form the **DV** denominator. We report retention per task, covering object counting, absolute and relative distance, sizes, direction, route planning, and appearance order. For **Latent Similarity**, we render 64 frames from

Table 1: **A subset of the BVB results**, with proprietary and open-weight models listed separately. **Rank** is the global ordering over all 51 configurations (Table 3). Colored effort badges encode the reasoning setting (red for `low`, amber for `medium`, green for `high` or `xhigh`, no badge for `none`). Darker and lighter purple mark the best and second-best among shown rows.

Model & Reasoning Effort	Rank	Overall	Dual VQA (DV) ↑									LS ↑		
			Obj. Count	Abs. Dist.	Obj. Size	Room Size	Rel. Dist.	Rel. Dir.	Route Plan	Appr. Order	**Avg.**	Layout	Motion	**Avg.**
Proprietary Models														
GPT-6-Astra high	1	70.07	48.5	33.5	73.7	33.1	49.0	50.0	54.8	36.0	53.7	87.3	90.0	88.6
GPT-5.6-Sol xhigh	2	67.49	38.0	39.9	69.0	53.1	43.6	50.7	47.3	16.0	51.7	84.0	86.7	85.4
Grok-4.6 xhigh	3	67.17	40.5	42.2	66.2	46.9	42.7	52.5	59.1	24.0	52.1	82.9	85.5	84.2
Qwen3.8-Max high	4	66.24	42.5	37.6	65.8	45.4	38.6	56.1	72.0	30.0	52.7	79.9	82.8	81.3
Claude-Opus-5 high	5	66.21	42.0	46.2	66.4	43.8	44.0	52.0	65.6	16.0	52.6	79.8	82.9	81.4
GPT-5.6-Sol high	6	65.73	34.5	37.6	65.0	40.8	37.8	52.2	63.4	26.0	49.8	82.5	85.3	83.9
GPT-5.6-Terra high	8	64.76	37.5	34.1	66.2	45.4	30.3	52.5	63.4	16.0	49.2	81.0	83.9	82.4
GPT-5.6-Sol	10	64.09	31.0	37.0	67.1	37.7	40.2	50.7	61.3	22.0	49.5	79.1	82.0	80.6
GPT-5.6-Luna high	13	63.58	28.5	36.4	65.2	34.6	37.3	54.4	54.8	10.0	48.2	79.6	82.7	81.1
Muse-Spark-1.3 xhigh	14	63.40	35.0	36.4	64.1	35.4	40.2	51.2	62.4	20.0	48.9	78.3	81.2	79.7
Grok-4.5 high	16	62.81	35.0	39.3	64.1	42.3	39.8	47.3	59.1	14.0	48.4	77.6	80.4	79.0
Gemini-3.1-Pro high	21	62.23	30.0	44.5	65.2	43.1	41.1	55.1	63.4	20.0	51.1	72.9	76.1	74.5
Gemini-3.8-Flash high	22	62.08	41.5	35.8	65.4	37.7	42.7	44.9	51.6	18.0	48.4	76.0	78.8	77.4
Claude-Sonnet-5 high	27	60.63	29.0	37.0	65.4	28.5	39.8	54.7	54.8	10.0	48.3	72.7	76.1	74.4
Claude-Sonnet-4.6 high	28	60.53	35.5	32.4	62.8	34.6	36.9	52.9	58.1	12.0	47.7	73.4	76.5	74.9
Claude-Opus-4.6	31	60.23	28.0	39.3	63.7	30.0	46.1	48.0	50.5	24.0	47.5	73.0	75.9	74.5
Gemini-3-Flash high	35	58.91	31.5	38.7	66.4	40.8	39.0	52.5	58.1	18.0	49.6	67.3	70.7	69.0
Claude-Opus-4.8 high	36	58.46	29.0	33.5	60.9	30.0	41.1	51.7	54.8	16.0	46.4	69.9	73.9	71.9
GPT-5.5	38	57.03	21.5	24.3	59.8	30.8	34.9	46.3	59.1	14.0	42.6	71.8	75.4	73.6
GLM-5V-Turbo	41	56.47	25.5	42.2	61.5	36.9	32.8	52.9	59.1	12.0	46.8	65.4	68.7	67.0
Grok-4.3 high	44	54.08	22.0	36.4	57.5	32.3	41.1	51.0	52.7	12.0	44.7	62.8	65.9	64.3
Seed-2.0-Mini	47	52.24	17.0	31.8	61.5	20.8	31.1	52.9	61.3	4.0	43.4	59.8	64.0	61.9
Qwen3.6-Plus	48	51.28	20.5	31.8	54.1	30.0	29.9	48.5	61.3	12.0	41.4	60.8	63.7	62.2
Seed-2.0-Lite	50	49.47	10.5	33.5	57.9	30.0	32.0	48.5	52.7	8.0	41.3	56.2	60.6	58.4
Open-weight Models														
GLM-5.3-Flash xhigh	12	63.96	35.0	38.2	64.5	46.2	46.9	51.7	54.8	28.0	50.8	77.0	80.3	78.7
Kimi-K2.5	42	55.91	24.0	32.9	59.2	33.8	36.9	47.5	58.1	6.0	44.0	67.7	70.8	69.2
MiniMax-M3	43	55.29	20.0	35.8	61.5	30.8	40.2	49.5	62.4	14.0	45.6	64.3	67.6	65.9
Qwen3.5-122B	45	53.50	16.5	32.9	63.9	25.4	29.0	52.9	57.0	8.0	44.1	62.0	65.6	63.8
Qwen3.5-397B	46	52.75	16.0	32.9	60.9	29.2	38.2	45.1	57.0	10.0	43.0	61.9	65.2	63.5
Qwen3.5-27B	51	47.50	13.5	27.2	54.5	26.2	30.7	45.3	49.5	4.0	38.6	55.8	58.9	57.3

each reconstruction with Blender's EEVEE renderer along the scene-camera timeline and sample 64 frames from the source video. The encoder is V-JEPA 2.1 ViT-G (Mur-Labadia et al., 2026).

3.2 Main Results

Overview. Table 1 shows a subset of the results. Appendix B lists all 51 configurations. Across all 51 configurations, **Overall** spans 47.50–70.07, **DV** spans 38.6–53.7, and **LS** spans 57.2–88.6. Better configurations tend to improve on both axes, but neither axis determines the other. Most importantly, the best **DV** is only 53.7. Even the best model therefore loses nearly half of the recoverable facts.

Rank and frontier. GPT-6-Astra-`high` leads at 70.07 **Overall**, followed by GPT-5.6-Sol-`xhigh` at 67.49 and Grok-4.6-`xhigh` at 67.17. The top-two **Overall** gap of 2.58 points is statistically significant (see Appendix C for bootstrap details), confirming that BVB separates even the strongest models. The corresponding difference in **DV** is not statistically significant. GLM-5.3-Flash-`xhigh` is the strongest open-weight configuration at rank 12 and 63.96 **Overall**, but the leading proprietary configurations still score higher on both axes. The best choice also depends on the budget (Section 4.5). Section 4 examines semantic failure, human alignment, and reasoning effort.

Finding 1. Multimodal agents can already understand video through programmatic reconstruction, but BVB is not saturated and separates models at every price tier.

Human blind ranking. Fifteen human raters ranked 5 anonymized reconstructions against the source video on nine scenes each, without knowing which model produced which reconstruction. Table 2 shows that their mean ranking matches the Overall order exactly (Spearman ρ=1.00). When measured per scene and per model, their preference correlates strongly with LS (Spearman ρ=0.83). Appendix I gives the full protocol and scene-level calibration.

Table 2: **Blind ranking matches the BVB Overall order** for all 5 selected configurations.

Model & Reasoning Effort	Human rank ↓	Mean rank ↓	BVB rank ↓
GPT-5.6-Sol xhigh	**1**	**1.47**	**2**
Grok-4.5 high	2	2.23	16
Gemini-3.1-Pro high	3	3.03	21
Claude-Opus-4.6	4	3.70	31
Qwen3.5-397B	5	4.58	46

4 Analysis

Section 3 ranks configurations by Overall, but a single score does not show which spatial and temporal facts are lost, whether the metrics match human judgment, or how reasoning effort and cost affect the results. We examine each of these questions below.

4.1 How large is the gap between looking right and being right?

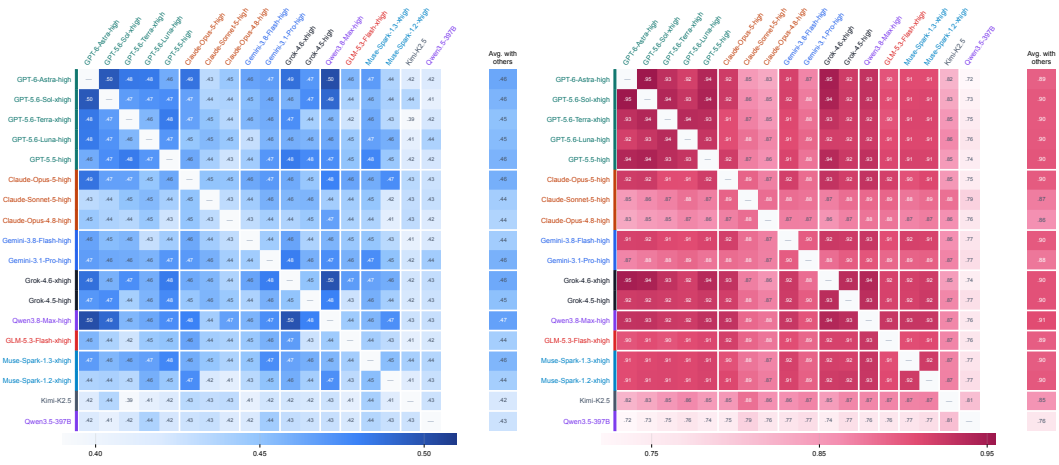
The best LS reaches 88.6, yet the best DV is only 53.7. In absolute terms, 981 of the 1,827 spatiotemporal questions that the VLM judge answers correctly on the source video are still answered correctly after reconstruction. The remaining 846 questions are lost even by the strongest model. High perceptual similarity from the frozen V-JEPA encoder therefore does not guarantee that the reconstruction preserves the spatial and temporal facts of the source.

Figure 3 compares reconstructions across agents. Across the 153 pairs formed by the 18 configurations, the mean V-JEPA similarity between rendered videos is 0.88, whereas the mean Jaccard overlap between the recoverable facts they retain is only 0.45. This overlap is still well above the 0.33 expected if each agent retained facts independently at its own rate, and every pair exceeds its expected value. Agents therefore share a common core of easily retained facts. Appearance also follows model family more closely than retained facts do. Same-family pairs average 0.91 in V-JEPA similarity, compared with 0.88 for cross-family pairs, while their Jaccard overlap rises only from 0.45 to 0.46. The most similar-looking pair, GPT-6-Astra-`high` and GPT-5.6-Sol-`xhigh`, belongs to one family, whereas the pair sharing the most retained facts, GPT-6-Astra-`high` and Qwen3.8-Max-`high`, spans two. Stronger agents also converge. Among the 18 configurations, pairs within the top nine by Overall average 0.93 in V-JEPA similarity and 0.47 in Jaccard overlap, compared with 0.86 and 0.44 within the bottom nine. Because most same-family pairs involve strong GPT models, part of the family effect reflects this convergence. Visual agreement between agents therefore overstates how much scene content they share.

Finding 2. Looking right is not the same as being right. The best model reaches 88.6 LS but retains only 53.7% of the recoverable facts, so nearly half of them are still lost after reconstruction.

4.2 What spatiotemporal information is retained after reconstruction?

Figure 4 summarizes task-level DV distributions across all configurations. Across all 51 configurations, object size and route planning have the highest mean retention at 63.0% and 58.5%, while appearance order and object count are lowest at 16.1% and 29.3%. The benchmark therefore exposes a shared hierarchy of task difficulty. Tasks that depend on more scene structure have lower retention. Object size is a property of a single object, and it has the highest retention. Object counting requires enumerating every instance in the room, and appearance order requires tracking the full camera trajectory. Both tasks lose most of their originally correct answers after reconstruction. The spread across models also varies by task. Object count spans 10.5–48.5% across configurations, room size spans 20.8–53.1%, and appearance order spans 4.0–36.0%. Relative direction is much more compressed at 44.9–57.4%. Table 1 further shows that no configuration dominates every task. Thus BVB contains both shared difficulty patterns and task-specific model rankings.



(a) Retained-answer overlap (b) V-JEPA similarity

Figure 3: **Reconstructions converge in appearance but diverge in semantic content.** (a) Jaccard overlap of the recoverable facts retained by each pair. (b) V-JEPA similarity between renders.

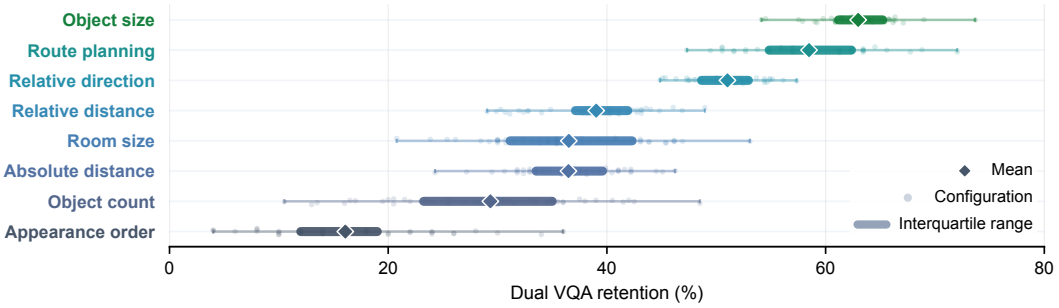


Figure 4: **Retention varies widely across tasks.** Full range, interquartile range, and mean **DV** over all 51 configurations for each task.

Finding 3. Task difficulty follows a consistent order across models, yet no single model is best on every task. Retention is highest for single-object properties such as size, and lowest for tasks that require reasoning over the whole scene, such as counting and appearance order.

4.3 Are the metrics complementary and human-aligned?

DV and **LS** improve together overall, but they rank models differently. Thus, we report both axes beside **Overall**. Figure 5 shows representative examples. Given the same source video, each model reconstructs different objects, different layouts, and different portions of the timeline. The human blind ranking from Section 3.2 lets us test whether the automatic metrics agree with human preferences. At the scene-model level, **LS** correlates with human preference at Spearman ρ=0.83 (Appendix Figure 11). **DV** is much weaker at ρ=0.11. Human rankings align more strongly with perceptual similarity than with semantic retention in our study. At the model level, **Overall** preserves the complete human ordering of the five tested configurations (Spearman ρ=1.00). The two metrics therefore serve different roles. **LS** tracks perceptual preference, while **DV** ensures that visually similar reconstructions do not receive a high score when they lose factual content. Appendix G examines the questions that the judge answers incorrectly on the source video.

Finding 4. **DV** and **LS** are related but distinct. Reconstructions reproduce appearance more reliably than factual content. **LS** tracks human preference, while **DV** measures semantic retention.

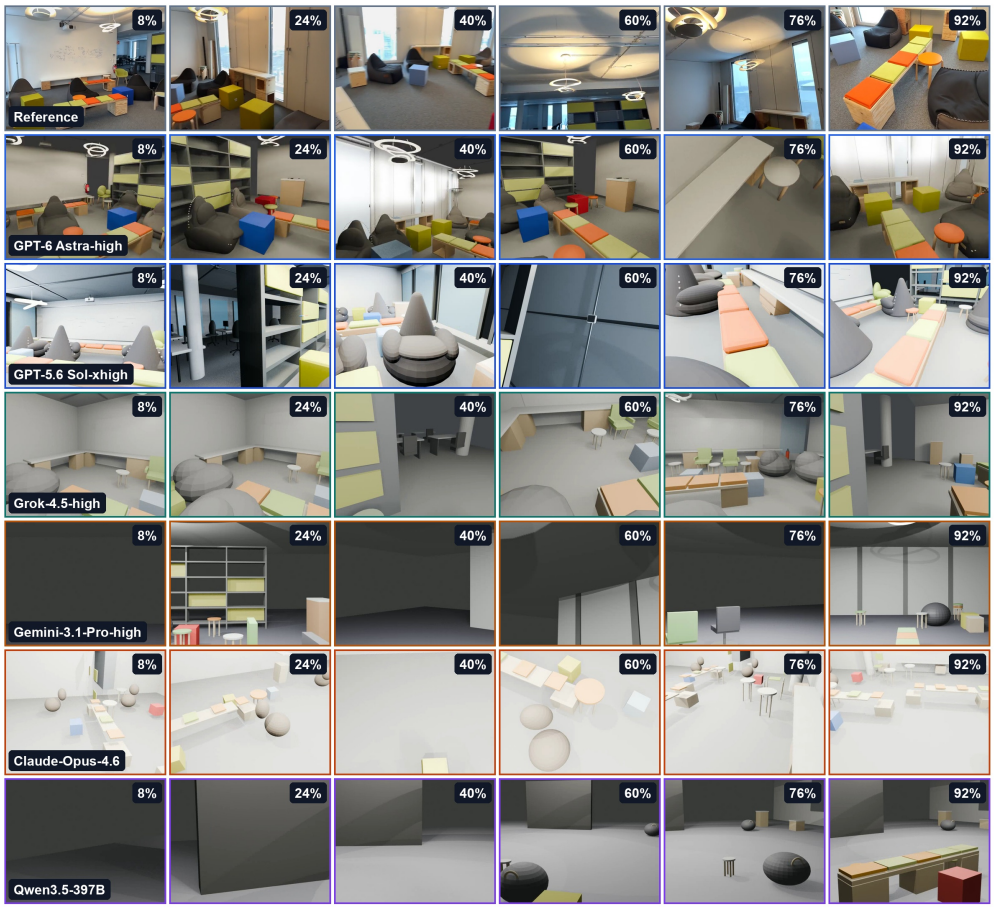


Figure 5: **Agents reconstruct the same video very differently.** Columns are matched video times.

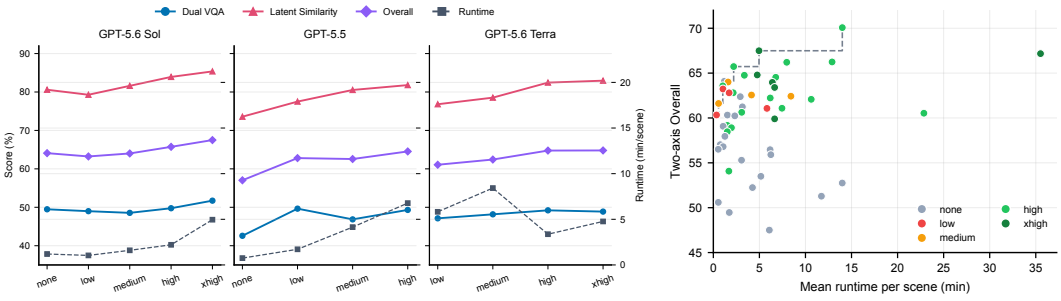
4.4 How do reasoning effort and runtime affect scores?

Provider reasoning controls do not change the two scores equally (Figure 6a). Across the GPT-5.6 Sol, GPT-5.5, and GPT-5.6 Terra ladders, **LS** generally rises with effort, whereas **DV** stays flat or even decreases. A likely explanation is that additional reasoning helps the model refine geometry, materials, and camera motion, all of which improve visual similarity, but does not lead the model to verify factual details such as object counts or spatial relations. **Overall** improves from the lowest to the highest available effort in each family, but the intermediate steps do not always increase. Runtime behaves similarly (Figure 6b). Slower configurations do not consistently score higher, and several fall below the **Overall** frontier set by faster ones. Extra inference time is therefore not a sufficient explanation for score differences and not a reliable indicator of reconstruction quality.

Finding 5. Across the tested effort ladders, higher reasoning effort generally improves **LS** more reliably than **DV**. Longer runtime does not mean higher **Overall**.

4.5 What does the frontier cost?

Mean Stage-1 spend ranges from $0.024 to $2.157 per scene, so the most expensive configuration costs roughly 90 times more than the cheapest (Figure 1). Spending more does not always improve quality. The top **Overall** configuration costs $1.258 per scene, while GPT-5.6-Sol-`xhigh` reaches 96% of that score at $0.778 and GLM-5.3-Flash-`xhigh` reaches 91% at $0.024. The cost difference is driven mainly by model family and reasoning effort level, not by scene difficulty. The **DV** and **LS** frontiers differ across price tiers (Appendix Figure 13), so the most cost-effective configurations depend on the evaluation axis. Evaluating one top-ranked configuration on all 288 scenes costs about $360, and most configurations cost much less, so evaluating a new model is affordable.



(a) Provider effort ladders (b) Runtime versus **Overall**

Figure 6: **Reasoning effort and runtime show different scaling behavior.** LS rises more consistently than DV across provider effort ladders, while **Overall** combines the two axes and still reverses at intermediate steps. Longer runtime alone does not mean higher **Overall**.

Finding 6. Higher cost does not guarantee proportional quality gains. Configurations at a fraction of the top price still reach over 90% of the best **Overall**.

5 Related Work

Benchmarking video understanding. Most video-understanding benchmarks evaluate models through question answering. VSI-Bench (Yang et al., 2025) probes visual-spatial intelligence in real indoor captures and finds that spatial reasoning remains a hard problem. EgoSchema (Mangalam et al., 2023) targets long-form video understanding, MMPerspective (Tang et al., 2025c) probes perspective understanding, and Eagle (Bi et al., 2024) targets egocentric video. Questions in this format can often be answered from static appearance alone, as single-frame models can match multi-frame ones (Lei et al., 2023). A second family asks agents to act on video rather than answer questions about it. VideoGUI (Lin et al., 2024) evaluates GUI automation from instructional video, VideoWebArena (Jang et al., 2025) evaluates web tasks that need video evidence, and ScreenSpot-Pro (Li et al., 2025) and GUIXplore (Sun et al., 2025) test screen grounding across applications. These benchmarks score task completion in 2D screen environments. BVB instead asks the agent to rebuild the video as an editable 3D scene whose render preserves its layout and dynamics.

Programs as visual representations. Writing code is an increasingly common intermediate step in visual reasoning because it produces a persistent artifact that can be executed and revised. ViperGPT (Surís et al., 2023) and VISPROG (Gupta & Kembhavi, 2023) compose perception modules into generated Python and read answers from execution traces. A second line generates programs that rebuild depicted content. SceneCraft (Hu et al., 2024) synthesizes 3D scenes from layout descriptions, VIGA (Yin et al., 2026) iteratively writes, renders, and revises Blender code, and Kubrick (He et al., 2025) coordinates multimodal agents that compose Blender scripts for synthetic video. BlenderGym (Gu et al., 2025) benchmarks graphics editing across placement, lighting, and geometry. EZBlender (Wang et al., 2026) decomposes editing instructions with a Plan-and-ReAct agent, and BlenderMCP (Ahuja, 2025) exposes Blender as a language model tool. Related systems address spatial variables (Chen et al., 2026), simulator code (Liang et al., 2026), vector animation (Yang et al., 2026b), and top-down room synthesis (Yang et al., 2026a). SceneActBench (Zhao et al., 2026) decomposes Blender tasks with hidden geometric ground truth, offering stronger per-task supervision. BVB instead targets holistic reconstruction from uncalibrated real video, supplies no assets, and relies on source-conditioned semantic and perceptual evaluation.

6 Conclusion

We propose BVB, which evaluates agentic video understanding by asking multimodal agents to reconstruct real-world videos in Blender. We evaluate 51 configurations across 10 model families, including GPT-6 Astra, and find that current agents produce visually plausible but semantically incomplete reconstructions. Our two-axis evaluation favors balanced performance across semantic retention and perceptual similarity, and we validate it against human preferences. Our results also show which video understanding abilities remain unsolved and where future work is most needed.

AI USE STATEMENT

We used generative AI tools to draft and revise prose under author direction, to implement evaluation and figure code from author-written specifications, and to assist with consistency checks across the manuscript and result artifacts. The authors selected the research questions, benchmark protocol, evidence, claims, and final wording, reviewed all AI-assisted output, and take responsibility for the paper. Language models also appear inside BVB itself, as the models under evaluation and as evaluator components, which Sections 2.2 and 3.1 document.

ETHICS STATEMENT

BVB is derived from the real indoor videos of VSI-Bench (Yang et al., 2025), whose captures come from ARKitScenes, ScanNet, and ScanNet++. We collect no new footage and redistribute no source video, so those clips remain governed by the consent, privacy, and license terms of the original datasets. Our own release, covering the harness, editable Blender artifacts, evaluation code, and per-scene scores, carries the MIT license of our code base. The blind ranking study of Appendix I distributed a self-contained form to individual raters, each identified only by an anonymous slot label and shown only video. It recorded no names, contact details, demographics, or other personal data, and the released scoring artifacts contain no rater-level identifiers. All agent runs ran against commercial inference APIs and open-weight checkpoints under their respective terms of service, and Appendix J reports the resulting spend.

REPRODUCIBILITY STATEMENT

Section 2.1 specifies the Stage-1 sandbox, its two agent actions, the conditions under which a run counts as valid, and the per-scene spend limit. Section 2.2 and Equation (1) define the two axes and the aggregate, derived in Appendix D. Appendices E and F record the judge, frozen encoder checkpoint, and frame counts. Appendix I gives the human-study protocol and states what it cannot establish, and Appendix B lists every scored configuration.

REFERENCES

Siddharth Ahuja. BlenderMCP: Connecting Blender to LLMs through the model context protocol. `https://github.com/ahujasid/blender-mcp`, 2025.

Mahmoud Assran, Adrien Bardes, David Fan, Quentin Garrido, Russell Howes, Mojtaba Komeili, Matthew Muckley, Ammar Rizvi, Claire Roberts, Koustuv Sinha, et al. V-JEPA 2: Self-supervised video models enable understanding, prediction and planning. *arXiv preprint arXiv:2506.09985*, 2025.

Adrien Bardes, Quentin Garrido, Jean Ponce, Xinlei Chen, Michael Rabbat, Yann LeCun, Mido Assran, and Nicolas Ballas. Revisiting feature prediction for learning visual representations from video. *Transactions on Machine Learning Research*, 2024.

Gilad Baruch, Zhuoyuan Chen, Afshin Dehghan, Tal Dimry, Yuri Feigin, Peter Fu, Thomas Gebauer, Brandon Joffe, Daniel Kurz, Arik Schwartz, and Elad Shulman. ARKitScenes: A diverse real-world dataset for 3d indoor scene understanding using mobile RGB-D data. In *Advances in Neural Information Processing Systems (NeurIPS), Datasets and Benchmarks Track*, 2021.

Jing Bi, Yunlong Tang, Luchuan Song, Ali Vosoughi, Nguyen Nguyen, and Chenliang Xu. EAGLE: Egocentric AGgregated language-video engine. In *Proceedings of the ACM International Conference on Multimedia (MM)*, 2024.

ByteDance Seed. Seed1.5-VL technical report. *arXiv preprint arXiv:2505.07062*, 2025.

Jieneng Chen, Wenxin Ma, Ruisheng Yuan, Yunzhi Zhang, Jiajun Wu, and Alan Yuille. Thinking with spatial code for physical-world video reasoning. *arXiv preprint arXiv:2603.05591*, 2026.

Angela Dai, Angel X. Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Nießner. ScanNet: Richly-annotated 3d reconstructions of indoor scenes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

Chaoyou Fu, Yuhan Dai, Yongdong Luo, Lei Li, Shuhuai Ren, Renrui Zhang, et al. Video-MME: The first-ever comprehensive evaluation benchmark of multi-modal LLMs in video analysis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025.

Gemini Team. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. *arXiv preprint arXiv:2507.06261*, 2025.

GLM-V Team. GLM-4.5V and GLM-4.1V-thinking: Towards versatile multimodal reasoning with scalable reinforcement learning. *arXiv preprint arXiv:2507.01006*, 2025.

Yunqi Gu, Ian Huang, Jihyeon Je, Guandao Yang, and Leonidas Guibas. BlenderGym: Benchmarking foundational model systems for graphics editing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 18574–18583, 2025.

Tanmay Gupta and Aniruddha Kembhavi. Visual programming: Compositional visual reasoning without training. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 14953–14962, 2023.

Liu He, Yizhi Song, Hejun Huang, Pinxin Liu, Yunlong Tang, Daniel Aliaga, and Xin Zhou. Kubrick: Multimodal agent collaborations for synthetic video generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, 2025.

Ziniu Hu, Ahmet Iscen, Aashi Jain, Thomas Kipf, Yisong Yue, David A Ross, Cordelia Schmid, and Alireza Fathi. SceneCraft: An LLM agent for synthesizing 3d scenes as blender code. In *International Conference on Machine Learning (ICML)*, 2024.

Lawrence Keunho Jang, Yinheng Li, Dan Zhao, Charles Ding, Justin Lin, Paul Pu Liang, Rogerio Bonatti, and Kazuhito Koishida. VideoWebArena: Evaluating long context multimodal agents with video understanding web tasks. In *International Conference on Learning Representations (ICLR)*, 2025.

Yunseok Jang, Yale Song, Youngjae Yu, Youngjin Kim, and Gunhee Kim. TGIF-QA: Toward spatio-temporal reasoning in visual question answering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

Kimi Team. Kimi K2: Open agentic intelligence. *arXiv preprint arXiv:2507.20534*, 2025.

Jie Lei, Licheng Yu, Mohit Bansal, and Tamara L. Berg. TVQA: Localized, compositional video question answering. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2018.

Jie Lei, Tamara L. Berg, and Mohit Bansal. Revealing single frame bias for video-and-language learning. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*, 2023.

Kaixin Li, Ziyang Meng, Hongzhan Lin, Ziyang Luo, Yuchen Tian, Jing Ma, Zhiyong Huang, and Tat-Seng Chua. ScreenSpot-Pro: GUI grounding for professional high-resolution computer use. In *ICLR Workshop on Reasoning and Planning for Large Language Models*, 2025.

Kunchang Li, Yali Wang, Yinan He, Yizhuo Li, Yi Wang, Yi Liu, Zun Wang, Jilan Xu, Guo Chen, Ping Luo, Limin Wang, and Yu Qiao. MVBench: A comprehensive multi-modal video understanding benchmark. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.

Jiarong Liang, Max Ku, Ka-Hei Hui, Ping Nie, and Wenhu Chen. VisPhyWorld: Probing physical reasoning via code-driven video reconstruction. *arXiv preprint arXiv:2602.13294*, 2026.

Kevin Qinghong Lin, Linjie Li, Difei Gao, Qinchen Wu, Mingyi Yan, Zhengyuan Yang, Lijuan Wang, and Mike Zheng Shou. VideoGUI: A benchmark for GUI automation from instructional videos. In *Advances in Neural Information Processing Systems (NeurIPS), Datasets and Benchmarks Track*, 2024.

Karttikeya Mangalam, Raiymbek Akshulakov, and Jitendra Malik. EgoSchema: A diagnostic benchmark for very long-form video language understanding. In *Advances in Neural Information Processing Systems (NeurIPS), Datasets and Benchmarks Track*, 2023.

Meta. Introducing Muse Spark 1.3, 2026. URL https://research.meta.ai/blog/introducing-muse-spark-1-3.

MiniMax. MiniMax-01: Scaling foundation models with lightning attention. *arXiv preprint arXiv:2501.08313*, 2025.

Lorenzo Mur-Labadia, Matthew Muckley, Amir Bar, Mido Assran, Koustuv Sinha, Mike Rabbat, Yann LeCun, Nicolas Ballas, and Adrien Bardes. V-JEPA 2.1: Unlocking dense features in video self-supervised learning. *arXiv preprint arXiv:2603.14482*, 2026.

OpenAI. GPT-6 Astra: A new generation of intelligence. https://openai.com/index/gpt-6-astra/, September 2026. Accessed September 7, 2026.

Viorica Pătrăucean, Lucas Smaira, Ankush Gupta, Adrià Recasens, et al. Perception test: A diagnostic benchmark for multimodal video models. In *Advances in Neural Information Processing Systems (NeurIPS), Datasets and Benchmarks Track*, 2023.

Qwen Team. Qwen2.5-VL technical report. *arXiv preprint arXiv:2502.13923*, 2025.

Thomas Ricouard. Architectural visualization with Astra. https://developers.openai.com/blog/architectural-visualization-with-astra, September 2026. Accessed September 7, 2026.

Yuchen Sun, Shanhui Zhao, Tao Yu, Hao Wen, Samith Va, Mengwei Xu, Yuanchun Li, and Chongyang Zhang. GUI-Xplore: Empowering generalizable GUI agents with one exploration. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 19477–19486, 2025.

Dídac Surís, Sachit Menon, and Carl Vondrick. ViperGPT: Visual inference via python execution for reasoning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 11888–11898, 2023.

Yunlong Tang, Jing Bi, Siting Xu, Luchuan Song, Susan Liang, Teng Wang, Daoan Zhang, Jie An, Jingyang Lin, Rongyi Zhu, Ali Vosoughi, Chao Huang, Zeliang Zhang, Pinxin Liu, Mingqian Feng, Feng Zheng, Jianguo Zhang, Ping Luo, Jiebo Luo, and Chenliang Xu. Video understanding with large language models: A survey. *IEEE Transactions on Circuits and Systems for Video Technology (TCSVT)*, 2025a.

Yunlong Tang, Junjia Guo, Hang Hua, Susan Liang, Mingqian Feng, Xinyang Li, Rui Mao, Chao Huang, Jing Bi, Zeliang Zhang, Pooyan Fazli, and Chenliang Xu. VidComposition: Can MLLMs analyze compositions in compiled videos? In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025b.

Yunlong Tang, Pinxin Liu, Zhangyun Tan, Mingqian Feng, Rui Mao, Chao Huang, Jing Bi, Yunzhong Xiao, Susan Liang, Hang Hua, Ali Vosoughi, Luchuan Song, Zeliang Zhang, and Chenliang Xu. MMPerspective: Do MLLMs understand perspective? a comprehensive benchmark for perspective perception, reasoning, and robustness. In *Advances in Neural Information Processing Systems (NeurIPS), Datasets and Benchmarks Track*, 2025c.

Hao Wang, Wenhui Zhu, Shao Tang, Zhipeng Wang, Xuanzhao Dong, Xin Li, Xiwen Chen, Ashish Bastola, Xinhao Huang, Yalin Wang, and Abolfazl Razi. EZBlender: Efficient 3d editing with plan-and-react agent. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV) Workshops*, pp. 1343–1352, 2026.

Haoning Wu, Dongxu Li, Bei Chen, and Junnan Li. LongVideoBench: A benchmark for long-context interleaved video-language understanding. In *Advances in Neural Information Processing Systems (NeurIPS), Datasets and Benchmarks Track*, 2024.

Junbin Xiao, Xindi Shang, Angela Yao, and Tat-Seng Chua. NExT-QA: Next phase of question-answering to explaining temporal actions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.

Jihan Yang, Shusheng Yang, Anjali W. Gupta, Rilyn Han, Li Fei-Fei, and Saining Xie. Thinking in space: How multimodal large language models see, remember, and recall spaces. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 10632–10643, 2025.

Yixuan Yang, Zhen Luo, Wanshui Gan, Jinkun Hao, Junru Lu, Jinghao Yan, Zhaoyang Lyu, and Xudong Xu. Code-as-room: Generating 3d rooms from top-down view images via agentic code synthesis. *arXiv preprint arXiv:2605.18451*, 2026a.

Yiying Yang, Wei Cheng, Sijin Chen, Honghao Fu, Xianfang Zeng, Yujun Cai, Gang Yu, and Xingjun Ma. OmniLottie: Generating vector animations via parameterized lottie tokens. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 39293–39303, 2026b.

Chandan Yeshwanth, Yueh-Cheng Liu, Matthias Nießner, and Angela Dai. ScanNet++: A high-fidelity dataset of 3D indoor scenes. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2023.

Shaofeng Yin, Jiaxin Ge, Zora Zhiruo Wang, Chenyang Wang, Xiuyu Li, Michael J. Black, Trevor Darrell, Angjoo Kanazawa, and Haiwen Feng. Vision-as-inverse-graphics agent via interleaved multimodal reasoning. *arXiv preprint arXiv:2601.11109*, 2026.

Zhou Yu, Dejing Xu, Jun Yu, Ting Yu, Zhou Zhao, Yueting Zhuang, and Dacheng Tao. ActivityNet-QA: A dataset for understanding complex web videos via question answering. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, 2019.

Z.ai. GLM-5.3-Flash model card, 2026. URL `https://huggingface.co/zai-org/GLM-5.3-Flash`.

Yifei Zhao, Xiangxin Zhou, Wenhao Yang, Jiaqi Tang, Pu Jian, Huanjin Yao, Jiarui Yao, Haowei Lin, Chunchao Guo, Zhuo Chen, Wenkai Lyu, Jianzhu Ma, Xueqian Wang, and Wenxi Zhu. SceneActBench: Can agents act on the 3d scenes they see? *arXiv preprint arXiv:2607.22393*, 2026.

A Stage-1 Harness and Agent Prompt

Section 2.1 describes the Stage-1 loop. The agent runs on the host and drives a fresh Blender 4.2 Docker container for each scene. At every turn it may execute a `bash` block or request video frames with a `frames` block. The harness sets no turn or frame budget. The only effort limit is a per-scene dollar budget computed from provider usage. When cumulative cost reaches the limit the harness terminates the loop immediately without an extra save turn; a low-budget warning at 25% remaining gives the model advance notice to save. A run finishes normally when the model has consumed at least one frame, saved `/workspace/output/result.blend`, and returned `DONE`. Evaluation never feeds information back into this loop.

System prompt. Every configuration receives the prompt below unchanged. Only the video and dollar limit vary by scene.

```
You are a video-understanding agent evaluated on reconstructing an indoor scene
from a video as Blender Python (bpy). You drive a Linux sandbox with Blender
installed.

Interface:
- ```bash block: the harness runs it in the sandbox and returns stdout/stderr.
  Run Blender headless as: blender_run --python /workspace/scratch/build.py
  (wraps `xvfb-run -a blender --background`).
- ```frames block: request video frames to look at. The harness extracts them
  and shows them to you as images on the next turn. Content is either explicit
  timestamps in seconds (e.g. `0, 4.5, 12, 30`) or `count=N` for N evenly-spaced
```

```
  frames (optionally with `start=` `end=` seconds). Look at as many frames as
  you need; request more whenever you want.
- When the scene is finished and saved, reply with the single word DONE.

Work in this order - do NOT skip looking at the video:
1. FIRST request frames with a ```frames block, on their own, and WAIT. Do not
   write any bpy or say DONE in the same message as a ```frames request - the
   frames you request are only shown to you on the next turn.
2. Look at the returned frames, then build the scene with ```bash.
3. Request more frames to check details whenever useful, then revise.
4. Only after you have actually seen frames and saved result.blend, say DONE.
A scene built without looking at any frame is a failure.

You decide how much effort to spend: how many frames, how many turns. The raw
video is also at /workspace/video/video.mp4 inside the sandbox.

Scene/output rules:
- The final scene MUST be saved to /workspace/output/result.blend.
- Build ALL geometry from basic primitives only (cube, plane, cylinder, cone,
  uv_sphere, torus) and assemblies of them. Group each object's parts in a
  Collection named after the object.
- No imported models, no sculpting/arbitrary meshes, no geometry nodes,
  particles, physics, or image textures. Materials = a single Principled BSDF
  with numeric values only.
- 1 Blender unit = 1 meter. Keep Unit Scale = 1.0. Use radians for rotations.
- Include at least one camera and one light.
- Reproduce object counts, sizes, positions, and spatial relationships as
  faithfully as you can from the video.

Camera trajectory / temporal reconstruction:
- Reconstruct camera motion as an ANIMATION, not a single static viewpoint.
- Insert keyframes for camera location and rotation_euler over time and set
  frame_start / frame_end to span the motion.
```

B Full Leaderboard

Table 3 reports all 51 configurations. **Overall** is the two-axis square-root mean in Equation (1), and ranks are global over this full set. The main text retains representative peak and default configurations plus all open-weight ones.

C Leaderboard Uncertainty

We quantify the gap between GPT-6-Astra-`high` and GPT-5.6-Sol-`xhigh` with a paired scene bootstrap. Each of 10,000 repetitions samples the 288 scene IDs with replacement, then recomputes DV from retained and original-correct question counts, LS from the scene mean, and **Overall** from the two resampled axes.

The **Overall** lead is 2.58 points with a 95% interval of $[0.72, 4.45]$. The LS advantage is likewise stable, while the 1.97-point DV difference has an interval that crosses zero. We therefore treat Astra as the **Overall** and LS leader but do not claim a statistically significant DV advantage over Sol-`xhigh`.

D Derivation of the Square-Root Mean

For configuration i, let the set of evaluation axes be $\mathcal{A} = \{\mathrm{DV}, \mathrm{LS}\}$. Writing $r_i^a = \sqrt{s_i^a}$ and $\bar{r}_i = |\mathcal{A}|^{-1} \sum_{a \in \mathcal{A}} r_i^a$, Equation (1) defines $\bar{s}_i = \bar{r}_i^2$. Using the population variance across axes,

$$\mathrm{Var}_{a \in \mathcal{A}}(r_i^a) = \frac{1}{|\mathcal{A}|} \sum_{a \in \mathcal{A}} s_i^a - \bar{s}_i, \tag{2}$$

$$\bar{s}_i = \frac{\mathrm{DV}_i + \mathrm{LS}_i + 2\sqrt{\mathrm{DV}_i \mathrm{LS}_i}}{4}. \tag{3}$$

Hence the square-root mean is the arithmetic mean corrected by $\mathrm{Var}(\sqrt{s_i^a})$. It lies between the geometric and arithmetic means, increases whenever either axis improves, and remains nonzero when only one axis is zero. As a protective measure, the implementation clips each axis score to

Table 3: Full BVB results over 51 configurations. Dual VQA reports retention on the judge-correct source subset, and Latent Similarity reports frozen V-JEPA layout and motion agreement. OK% is the percentage of scenes that produced a renderable file. Darker and lighter purple mark the best and second-best values.

Model & Reasoning Effort	Rank	Overall	Dual VQA (DV) ↑									LS ↑			OK%
			Obj. Count	Abs. Dist.	Obj. Size	Room Size	Rel. Dist.	Rel. Dir.	Route Plan	Appr. Order	Avg.	Layout	Motion	Avg.	
Proprietary Models															
GPT-6-Astra high	1	70.07	48.5	33.5	73.7	33.1	49.0	50.0	54.8	36.0	53.7	87.3	90.0	88.6	100.0
GPT-5.6-Sol xhigh	2	67.49	38.0	39.9	69.0	53.1	43.6	50.7	47.3	16.0	51.7	84.0	86.7	85.4	98.6
Grok-4.6 xhigh	3	67.17	40.5	42.2	66.2	46.9	42.7	52.5	59.1	24.0	52.1	82.9	85.5	84.2	97.9
Qwen3.8-Max high	4	66.24	42.5	37.6	65.8	45.4	38.6	56.1	72.0	30.0	52.7	79.9	82.8	81.3	99.3
Claude-Opus-5 high	5	66.21	42.0	46.2	66.4	43.8	44.0	52.0	65.6	16.0	52.6	79.8	82.9	81.4	100.0
GPT-5.6-Sol high	6	65.73	34.5	37.6	65.0	40.8	37.8	52.2	63.4	26.0	49.8	82.5	85.3	83.9	99.7
GPT-5.6-Terra xhigh	7	64.79	33.0	30.6	65.8	43.1	40.2	50.7	62.4	12.0	48.9	81.6	84.3	82.9	97.9
GPT-5.6-Terra high	8	64.76	37.5	34.1	66.2	45.4	30.3	52.5	63.4	16.0	49.2	81.0	83.9	82.4	99.7
GPT-5.5 high	9	64.53	34.5	34.7	65.8	46.2	37.8	50.5	63.4	12.0	49.3	80.4	83.2	81.8	97.2
GPT-5.6-Sol	10	64.09	31.0	37.0	67.1	37.7	40.2	50.7	61.3	22.0	49.5	79.1	82.0	80.6	99.7
GPT-5.6-Sol medium	11	64.01	35.0	40.5	64.3	33.8	41.1	48.0	61.3	18.0	48.5	80.1	83.1	81.6	99.7
GPT-5.6-Luna high	13	63.58	28.5	36.4	65.2	34.6	37.3	54.4	54.8	10.0	48.2	79.6	82.7	81.1	99.0
Muse-Spark-1.3 xhigh	14	63.40	35.0	36.4	64.1	35.4	40.2	51.2	62.4	20.0	48.9	78.3	81.2	79.7	99.0
GPT-5.6-Sol low	15	63.23	28.5	41.0	63.0	35.4	39.4	54.4	62.4	22.0	49.0	77.8	80.8	79.3	99.3
Grok-4.5 high	16	62.81	35.0	39.3	64.1	42.3	39.8	47.3	59.1	14.0	48.4	77.6	80.4	79.0	96.5
GPT-5.5 low	17	62.80	35.0	33.5	61.8	42.3	44.8	54.4	55.9	26.0	49.6	76.0	79.0	77.5	99.0
GPT-5.5 medium	18	62.56	25.5	35.8	63.9	41.5	37.8	50.0	51.6	12.0	46.9	79.1	81.9	80.5	97.2
GPT-5.6-Terra medium	19	62.43	36.0	34.1	62.4	39.2	43.2	47.3	55.9	34.0	48.2	77.0	80.1	78.5	100.0
Claude-Sonnet-4.6	20	62.37	30.5	37.0	63.9	43.8	43.2	57.4	62.4	14.0	50.6	73.8	76.9	75.3	99.7
Gemini-3.1-Pro high	21	62.23	30.0	44.5	65.2	43.1	41.1	55.1	63.4	20.0	51.1	72.9	76.1	74.5	97.9
Gemini-3.8-Flash high	22	62.08	41.5	35.8	65.4	37.7	42.7	44.9	51.6	18.0	48.4	76.0	78.8	77.4	95.1
GPT-5.6-Luna medium	23	61.62	23.0	31.8	62.2	35.4	41.9	52.5	53.8	14.0	46.5	77.2	80.5	78.8	100.0
Claude-Sonnet-5	24	61.24	30.0	39.9	62.4	36.2	40.7	54.9	64.5	18.0	49.2	72.9	76.3	74.6	99.7
GPT-5.2 high	25	61.08	39.0	38.2	60.9	46.2	32.4	51.2	59.1	12.0	47.9	74.4	77.2	75.8	99.0
GPT-5.6-Terra low	26	61.07	27.0	36.4	62.4	38.5	40.2	49.0	57.0	24.0	47.1	75.2	78.4	76.8	99.7
Claude-Sonnet-5 high	27	60.63	29.0	37.0	65.4	28.5	39.8	54.7	54.8	10.0	48.3	72.7	76.1	74.4	100.0
Claude-Sonnet-4.6 high	28	60.53	35.5	32.4	62.8	34.6	36.9	52.9	58.1	12.0	47.7	73.4	76.5	74.9	97.9
GPT-5.6-Luna low	29	60.33	20.5	35.8	60.5	34.6	43.2	51.2	67.7	18.0	46.8	73.9	77.3	75.6	100.0
Claude-Opus-4.7	30	60.32	36.0	41.0	61.5	34.6	42.3	54.9	57.0	8.0	49.2	70.9	74.3	72.6	97.6
Claude-Opus-4.6	31	60.23	28.0	39.3	63.7	30.0	46.1	48.0	50.5	24.0	47.5	73.0	75.9	74.5	99.0
Muse-Spark-1.2 xhigh	32	59.90	33.5	32.4	62.8	36.9	34.0	47.5	62.4	10.0	46.2	74.1	76.7	75.4	93.8
Claude-Opus-4.7 high	33	59.17	33.5	42.2	61.5	35.4	38.2	53.7	55.9	14.0	48.3	69.3	72.9	71.1	96.9
Claude-Opus-4.8	34	59.07	30.5	37.0	62.0	35.4	41.9	48.8	57.0	18.0	47.2	70.4	74.1	72.2	100.0
Gemini-3-Flash high	35	58.91	31.5	38.7	66.4	40.8	39.0	52.5	58.1	18.0	49.6	67.3	70.7	69.0	100.0
Claude-Opus-4.8 high	36	58.46	29.0	33.5	60.9	30.0	41.1	51.7	54.8	16.0	46.4	69.9	73.9	71.9	100.0
GPT-5.2	37	57.95	30.0	41.0	60.9	31.5	40.2	50.7	50.5	12.0	46.7	68.7	72.2	70.4	99.7
GPT-5.5	38	57.03	21.5	24.3	59.8	30.8	34.9	46.3	59.1	14.0	42.6	71.8	75.4	73.6	99.3
GPT-5.4	39	56.80	23.5	41.6	64.8	43.1	32.8	48.5	50.5	16.0	46.6	66.3	69.6	68.0	97.6
GPT-5.4-Mini	40	56.51	19.5	45.1	60.9	23.8	40.7	52.2	63.4	14.0	46.5	65.7	69.4	67.5	100.0
GLM-5V-Turbo	41	56.47	25.5	42.2	61.5	36.9	32.8	52.9	59.1	12.0	46.8	65.4	68.7	67.0	94.1
Grok-4.3 high	44	54.08	22.0	36.4	57.5	32.3	41.1	51.0	52.7	12.0	44.7	62.8	65.9	64.3	94.4
Seed-2.0-Mini	47	52.24	17.0	31.8	61.5	20.8	31.1	52.9	61.3	4.0	43.4	59.8	64.0	61.9	97.2
Qwen3.6-Plus	48	51.28	20.5	31.8	54.1	30.0	29.9	48.5	61.3	12.0	41.4	60.8	63.7	62.2	88.5
Grok-4.3	49	50.59	13.0	29.5	59.0	30.0	40.2	52.5	68.8	12.0	44.4	55.4	59.0	57.2	99.0
Seed-2.0-Lite	50	49.47	10.5	33.5	57.9	30.0	32.0	48.5	52.7	8.0	41.3	56.2	60.6	58.4	92.7
Open-weight Models															
GLM-5.3-Flash xhigh	12	63.96	35.0	38.2	64.5	46.2	46.9	51.7	54.8	28.0	50.8	77.0	80.3	78.7	99.3
Kimi-K2.5	42	55.91	24.0	32.9	59.2	33.8	36.9	47.5	58.1	6.0	44.0	67.7	70.8	69.2	95.8
MiniMax-M3	43	55.29	20.0	35.8	61.5	30.8	40.2	49.5	62.4	14.0	45.6	64.3	67.6	65.9	97.6
Qwen3.5-122B	45	53.50	16.5	32.9	63.9	25.4	29.0	52.9	57.0	8.0	44.1	62.0	65.6	63.8	97.6
Qwen3.5-397B	46	52.75	16.0	32.9	60.9	29.2	38.2	45.1	57.0	10.0	43.0	61.9	65.2	63.5	97.2
Qwen3.5-27B	51	47.50	13.5	27.2	54.5	26.2	30.7	45.3	49.5	4.0	38.6	55.8	58.9	57.3	84.7

Table 4: Paired scene-bootstrap intervals for the top two configurations. Brackets give 95% percentile intervals, and the final column is the fraction of bootstrap repetitions with a positive Astra-minus-Sol difference.

Metric	Astra high	Sol xhigh	Difference	$\Pr_{\text{boot}}(\Delta > 0)$
Dual VQA	53.69 [51.24, 56.18]	51.72 [49.31, 54.12]	+1.97 [-0.91, +4.79]	0.903
Latent Similarity	88.62 [88.32, 88.92]	85.36 [84.00, 86.46]	+3.27 [+2.20, +4.60]	1.000
Overall	70.07 [68.65, 71.51]	67.49 [65.78, 69.10]	+2.58 [+0.72, +4.45]	0.997

$\max(0,\, s_i^a)$ before taking the square root; no evaluated configuration produces a negative score. The choice does not determine the headline ordering. GPT-6 Astra, Sol-xhigh, and Grok-4.6-xhigh remain the top three under arithmetic and geometric aggregation, and the full 51-model ranks correlate with the square-root ranks at ρ=0.999 and 0.998, respectively.

E Dual VQA Protocol

Dual VQA uses the judge `gpt-5.4-mini`. For every VSI-Bench question q, we sample 16 frames from the original video and 16 from the reconstructed camera render, ask the same shortest-answer prompt, and score both answers against the VSI-Bench target. Original-video answers are queried once and shared across every model run. The prompt instructs the judge to use only the provided frames and return a number, short phrase, or option letter without explanation. We set `max_completion_tokens=32` and `temperature=0`, and retry transient API failures up to six times. All reported runs use the same `gpt-5.4-mini` API alias.

Let o_q and $r_{i,q}$ indicate whether the original and reconstruction answers are correct for configuration i. We report conditional retention

$$\mathrm{DV}_i = \frac{\sum_q o_q r_{i,q}}{\sum_q o_q}. \tag{4}$$

The denominator contains the 1,827 questions the judge answers correctly from the original clips, out of 5,130 total questions. The judge-correct denominators for object count, absolute distance, object size, room size, relative distance, relative direction, route planning, and appearance order are respectively 200, 173, 532, 130, 241, 408, 93, and 50. This control prevents judge failures on the source video from being attributed to a reconstruction. Because reconstructions consist of untextured Blender primitives whose appearance differs from real video, the judge may find reconstruction frames harder to parse. A lower judge accuracy on reconstruction frames therefore reflects information lost during reconstruction, not a flaw in the evaluation. Results are also reported by VSI-Bench task, with relative-direction easy, medium, and hard subsets micro-aggregated into one column.

F Latent Similarity with V-JEPA 2.1

BVB compares each original video and camera render with a frozen V-JEPA 2.1 ViT-G encoder (`apiantonio/vjepa2.1-vit-gigantic-384`) (Bardes et al., 2024; Assran et al., 2025; Mur-Labadia et al., 2026). The encoder is never fine-tuned on BVB. This makes the metric independent of the evaluated agent and avoids training a scoring head on benchmark reconstructions.

Paired clips. We render T=64 RGB frames uniformly across the reconstruction's Blender camera timeline with EEVEE at 512-pixel resolution and sample 64 frames uniformly from the original mp4. Both clips cover their full temporal extent without assuming framewise registration.

Encoding and scores. Let $z_{\text{orig}}, z_{\text{rend}} \in \mathbb{R}^{N \times D}$ denote the last-layer token sequences. We average all tokens for a global clip representation and reshape tokens onto a $T_g \times H \times W$ grid for a temporally pooled layout map. With 64 input frames, patch size 16, and tubelet size 2, the ViT-G encoder

produces a $32 \times 32 \times 32$ token grid (N=32,768 tokens of dimension D=1664):

$$\bar{z}_c^{\text{mot}} = \text{mean}_{i=1}^{N}(z_{c,i}), \tag{5}$$

$$\bar{z}_c^{\text{lay}} = \text{mean}_{t=1}^{T_g}\big(\text{reshape}(z_c; T_g, H, W, D)_{t,:,:,:}\big), \tag{6}$$

$$\text{motion_sim} = \cos\left(\bar{z}_{\text{orig}}^{\text{mot}}, \bar{z}_{\text{rend}}^{\text{mot}}\right), \tag{7}$$

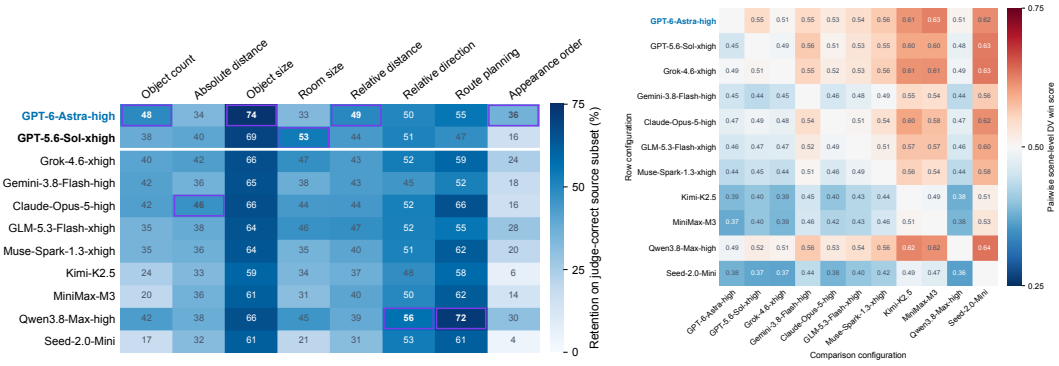
$$\text{layout_sim} = \frac{1}{HW}\sum_{h,w}\cos\left(\bar{z}_{\text{orig}}^{\text{lay}}[h,w], \bar{z}_{\text{rend}}^{\text{lay}}[h,w]\right), \tag{8}$$

$$\text{LS} = \tfrac{1}{2}\big(\text{layout_sim} + \text{motion_sim}\big). \tag{9}$$

Features are discarded after scoring. The release retains per-scene JSONL records and one run-level summary.

G Additional Score Diagnostics

These diagnostics use the same fixed source-correct question set, missing output handling, and per-scene LS records as the leaderboard. They show variation that a single run-level aggregate does not capture. Figure 7 enlarges the task profile from the main text and reports scene-level pairwise wins. Figures 8–9 then compare source collections and paired score differences.



(a) Dual VQA retention by task

(b) Pairwise scene-level DV wins

Figure 7: **Dual VQA difficulty is structured, and aggregate ranks do not reflect scene-level ties.** (a) Object size and route planning are retained more reliably than appearance order and object count. Purple boxes mark the best shown value in each task. (b) Each off-diagonal cell is the row configuration's scene-level DV win score against the column configuration over the same 278 scenes. Ties contribute one half.

Panel (b) of Figure 7 gives the top pair a scene-level win score of 0.55, considerably less decisive than a rank alone suggests. This agrees with the paired-bootstrap result that the top-two DV difference is not statistically significant. Table 5 gives the exact distribution summary behind the task-level plots.

Table 5: **Task difficulty and discrimination differ across BVB tasks.** Statistics summarize task-level DV over all 51 configurations. Object size has the highest mean, while object count has the widest range.

Task	Mean	Interquartile range	Full range
Object count	29.3	$[23.2, 35.0]$	$[10.5, 48.5]$
Absolute distance	36.5	$[33.5, 39.6]$	$[24.3, 46.2]$
Object size	63.0	$[61.2, 65.2]$	$[54.1, 73.7]$
Room size	36.5	$[31.2, 42.3]$	$[20.8, 53.1]$
Relative distance	39.0	$[37.1, 41.9]$	$[29.0, 49.0]$
Relative direction	51.0	$[48.7, 52.9]$	$[44.9, 57.4]$
Route planning	58.5	$[54.8, 62.4]$	$[47.3, 72.0]$
Appearance order	16.1	$[12.0, 19.0]$	$[4.0, 36.0]$

Recovery on judge-failed source questions. Retention conditions on the questions the judge answers correctly from the source, and we examine recovery on questions outside this subset. Restricted to scenes with a successful render, the judge recovers on average 22.6% of the judge-failed multiple-choice questions from the render, and no configuration exceeds 25.1%. Both values sit below the 27.9% chance level obtained by blending the two-, three-, and four-option formats over this question pool. Recovery on these questions remains below the nominal chance baseline across all evaluated configurations, indicating that the judge's errors on these questions are systematic rather than random and that the reconstruction does not provide additional signal to correct them. On the numeric tasks the recovery rate instead ranges from 14.5% to 19.9% and correlates with retention at Spearman 0.60 across the 51 configurations.

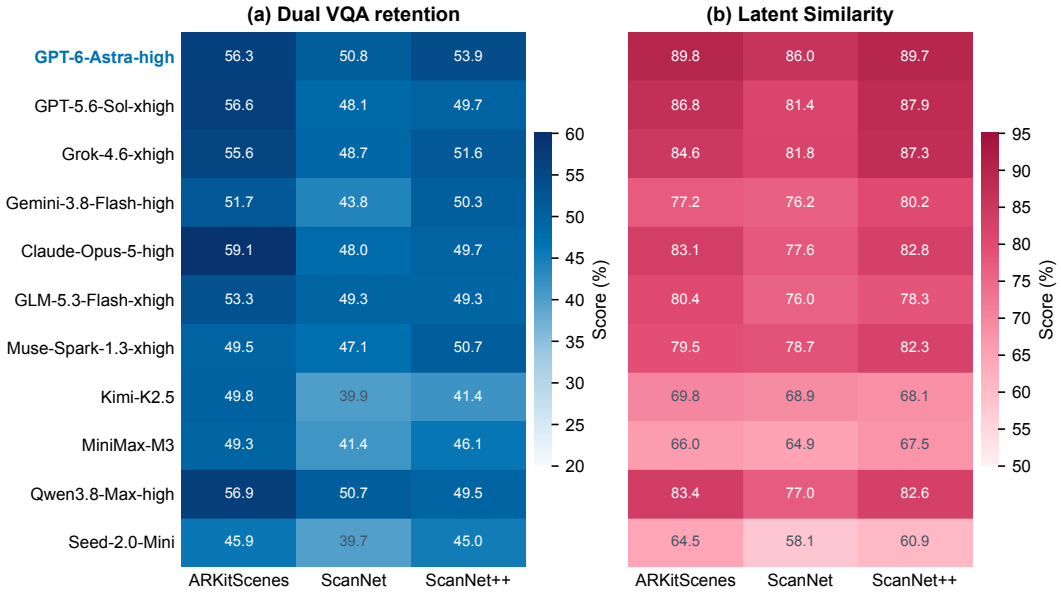


Figure 8: **Source collection shifts both evaluation axes.** Every shown configuration scores higher on ARKitScenes than ScanNet for both DV and LS. DV is micro-averaged over source-correct questions within each collection, while LS is averaged over its scene records.

The source profile in Figure 8 shows a domain effect shared across otherwise different configurations. Because the collections differ in capture style, scene composition, and question mix, these differences show sensitivity to the source domain rather than any single dataset factor.

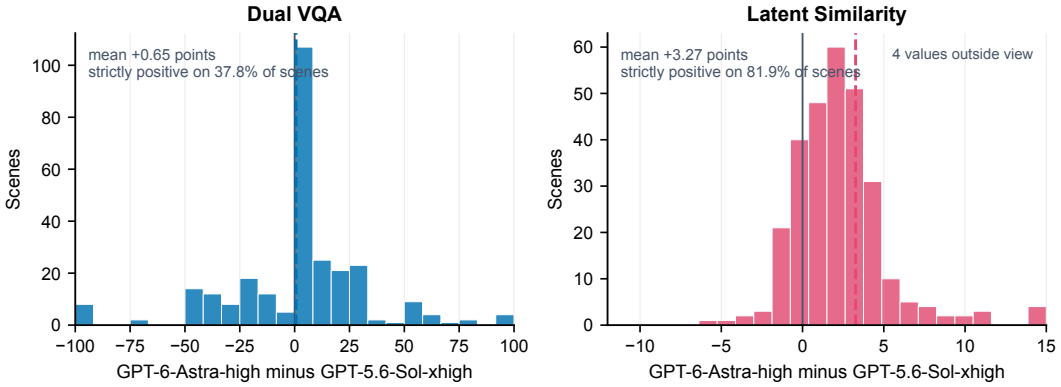


Figure 9: **Top-line gains are not uniform across scenes.** DV contains many identical scores and both positive and negative scene differences. LS is positive on most scenes. Four LS values beyond the displayed range correspond to Sol render failures scored as zero under the leaderboard policy.

The unweighted mean scene-level DV difference in Figure 9 is $+0.65$ points, whereas the pooled question-level leaderboard difference is $+1.97$ points. Scene-level DV denominators vary with the

number of source-correct questions, so the pooled score and paired scene summary answer different questions.

H Effort and Runtime Details

Table 6 reports how consistently each metric changes as provider-defined reasoning effort increases.

Table 6: **Latent Similarity responds most monotonically to effort.** Entries are within-family Spearman correlations between ordered effort settings and each score.

Model family	Settings	DV	LS	Overall
GPT-5.5	4	+0.40	+1.00	+0.80
GPT-5.6 Luna	3	+0.50	+1.00	+1.00
GPT-5.6 Sol	5	+0.60	+0.90	+0.70
GPT-5.6 Terra	4	+0.80	+1.00	+1.00
Mean		**+0.57**	**+0.97**	**+0.88**

Figure 10 separates the runtime and score change for each adjacent effort step. Figure 11 reports the human calibration used in the ranking study below.

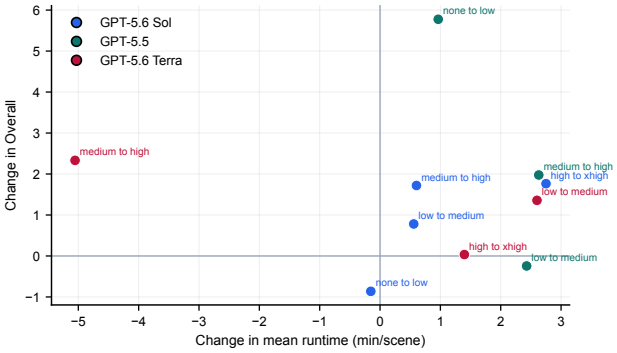


Figure 10: **Additional runtime is neither necessary nor sufficient for a Overall gain.** Each point compares adjacent effort settings within one provider family.

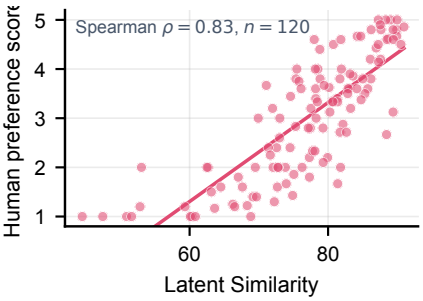


Figure 11: **Latent Similarity tracks blind visual preference.** Each point is one scene-model pair from the five-model study (Spearman ρ=0.83).

The pattern supports treating effort as a provider-specific control rather than a common compute scale. It also explains why the main-text runtime scatter does not form a single quality curve.

I Human Blind Ranking Study

Fifteen raters each saw nine of a fixed pool of 24 scenes balanced across ARKitScenes, ScanNet, and ScanNet++. For every scene, raters viewed the source clip beside five anonymized reconstructions and ranked them on object identity, spatial layout, camera path, and appearance order. The resulting 135 judgments per configuration yield mean ranks of 1.47, 2.23, 3.03, 3.70, and 4.58. Table 2 reports these model-level results in the main text.

At the scene-model level, Figure 11 shows that human preference correlates with Latent Similarity at Spearman ρ=0.83 over 120 observations. Dual VQA has lower correlation (ρ=0.11, n=110), consistent with its different role as a semantic retention measure. Its sample is smaller because 10 scene-model cases contain no judge-correct source question and therefore have no defined conditional retention. These observations are clustered within 24 scenes and five configurations, so the correlations are descriptive rather than based on 120 independent samples. The study supports the perceptual metric and the ordering of the five sampled configurations. The gap reflects a difference in task format. Ranking five reconstructions side by side is an inherently visual comparison, so raters weight overall appearance even when the criteria name factual elements. Dual VQA scores discrete questions independently and captures a different axis of reconstruction quality.

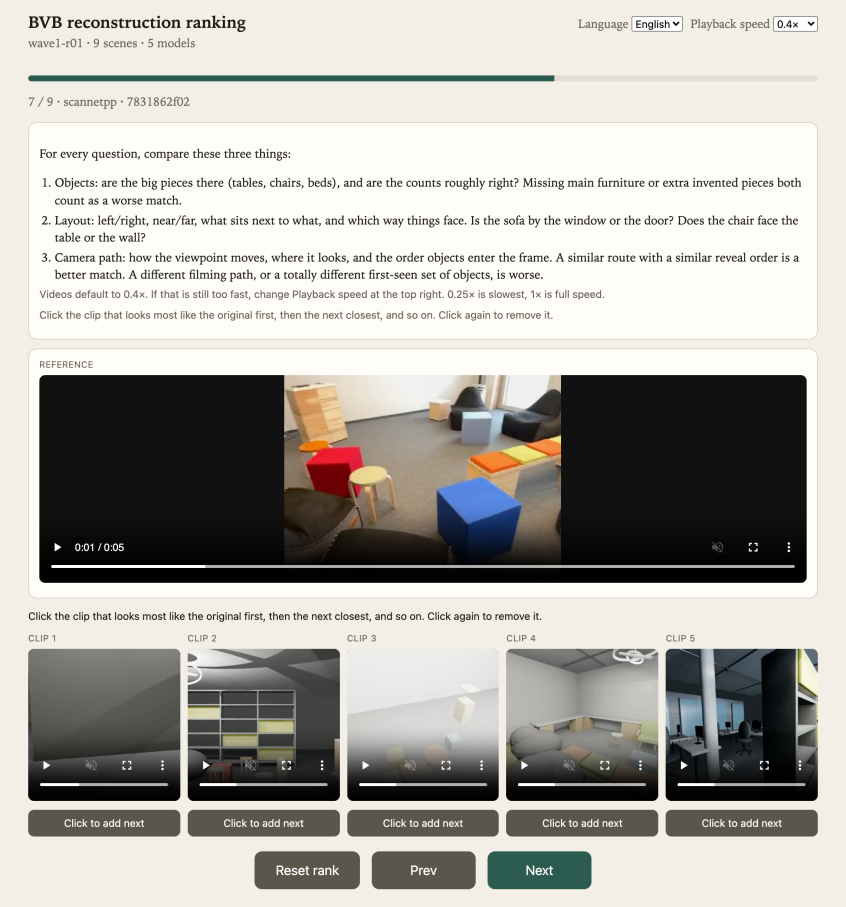


Figure 12: **Blind-ranking interface.** The reference video appears above five anonymized candidate reconstructions. Model identities were hidden, candidate order was randomized, and the form provided English and Chinese instructions.

J Cost Frontier

For each configuration, agent spend is the mean Stage-1 API cost per scene from provider usage accounting. It excludes the Dual VQA judge and frozen V-JEPA encoder, so it measures the cost of producing the reconstruction, not of scoring it. Across 51 configurations, spend ranges from $0.024 to $2.157 per scene.

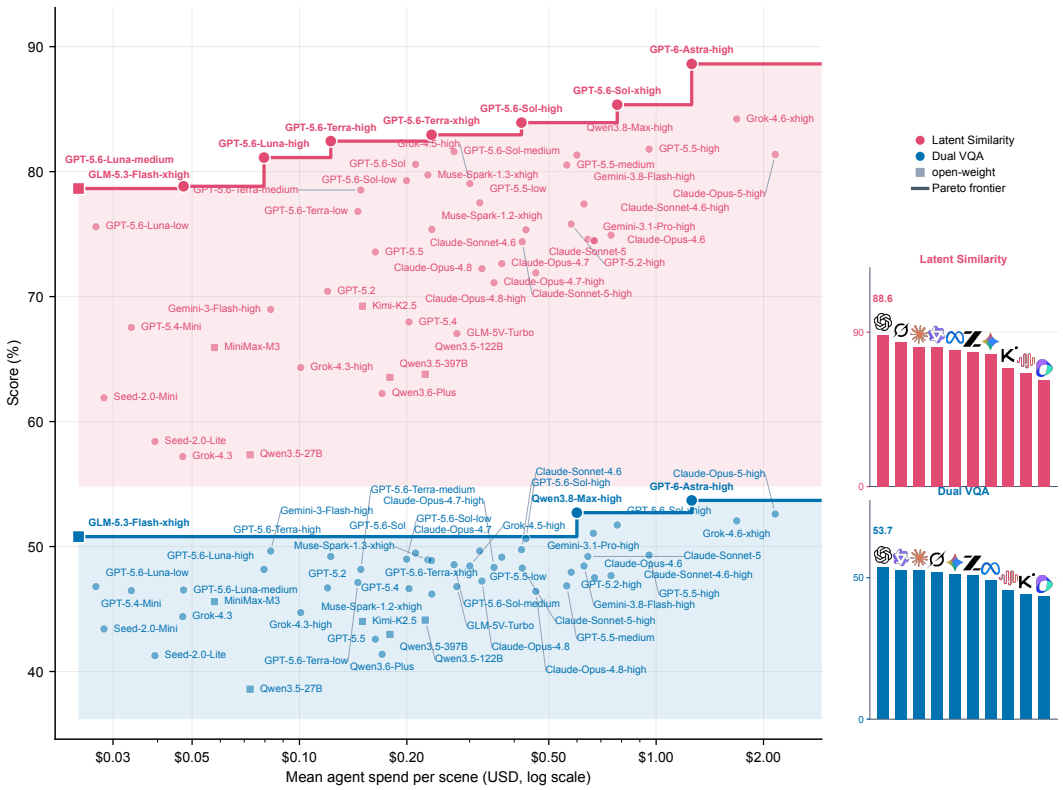


Figure 13: **Semantic and perceptual quality have different cost frontiers.** Each staircase contains configurations that no cheaper run outperforms on that axis, and the right panels show the best configuration from each model family.

Three configurations lie on the DV frontier and eight on the LS frontier. Both start with GLM-5.3-Flash and end with GPT-6 Astra; the DV frontier passes through Qwen3.8-Max, while the LS frontier includes Luna, Terra, and Sol. Astra reaches 53.7 **DV** and 88.6 **LS** at $1.258 per scene. Sol-`xhigh` reaches 51.7 and 85.4 at $0.778, while GLM-5.3-Flash-`xhigh` reaches 50.8 and 78.7 at $0.024. GLM-5.3-Flash is the only open-weight configuration on either measured frontier.

K Limitations

BVB covers indoor egocentric videos from three capture sources, and broader environments, outdoor scenes, and interactive editing remain future work. Performance also depends on coding ability and familiarity with Blender, so BVB evaluates video understanding through an agent's ability to express it programmatically. Both axes evaluate the rendered video, not the underlying 3D geometry, which matches the source-conditioned setting where aligned ground-truth geometry is unavailable. As in other judge-based evaluations, Dual VQA depends on a VLM judge. Conditioning on the judge-correct source subset and keeping the judge fixed across all configurations mitigate this dependence. The blind ranking study covers five configurations, and extending it to newer models is straightforward under the released protocol. Finally, the shared per-scene cost ceiling keeps comparisons uniform and affordable, though individual configurations might improve further under larger budgets.

L Qualitative Reconstruction Cases

The full candidate pool contains four scenes from each source. Figure 5 shows Candidate 12 in the main text, and the remaining eleven appear below. Every row uses the same six normalized clip times. From top to bottom, each candidate shows the reference, GPT-6 Astra, GPT-5.6 Sol, Grok,

Gemini, Claude, and Qwen. The five non-Astra configurations are exactly those used in the human blind study. These candidates support visual selection and do not estimate failure prevalence.

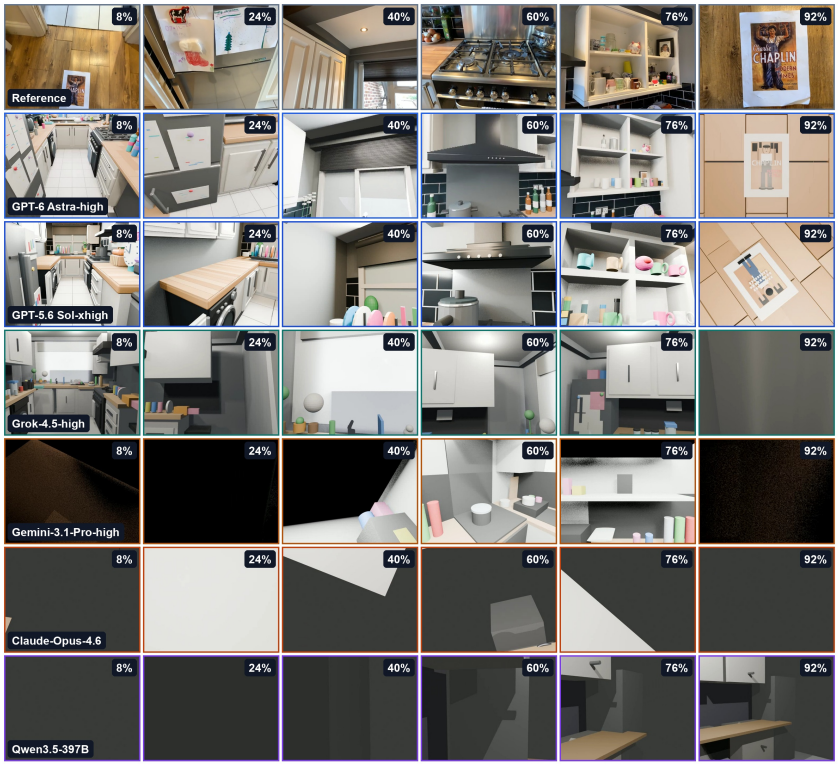


Figure 14: **Candidate 1: cluttered kitchen.** ARKitScenes scene 42446049 at six matched timestamps.

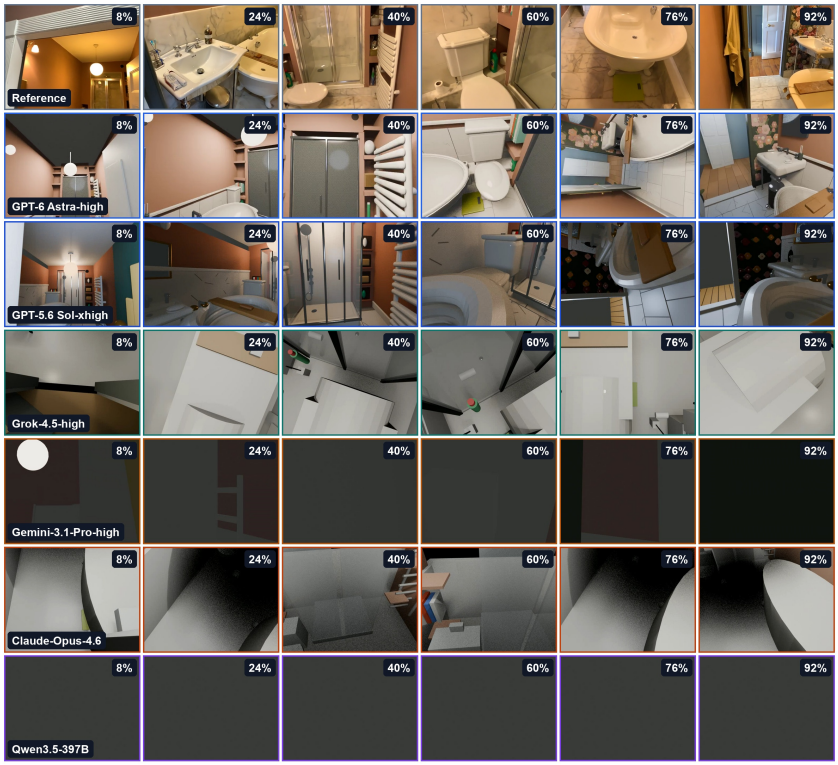


Figure 15: **Candidate 2: compact furnished room.** ARKitScenes scene 45260900 at six matched timestamps.

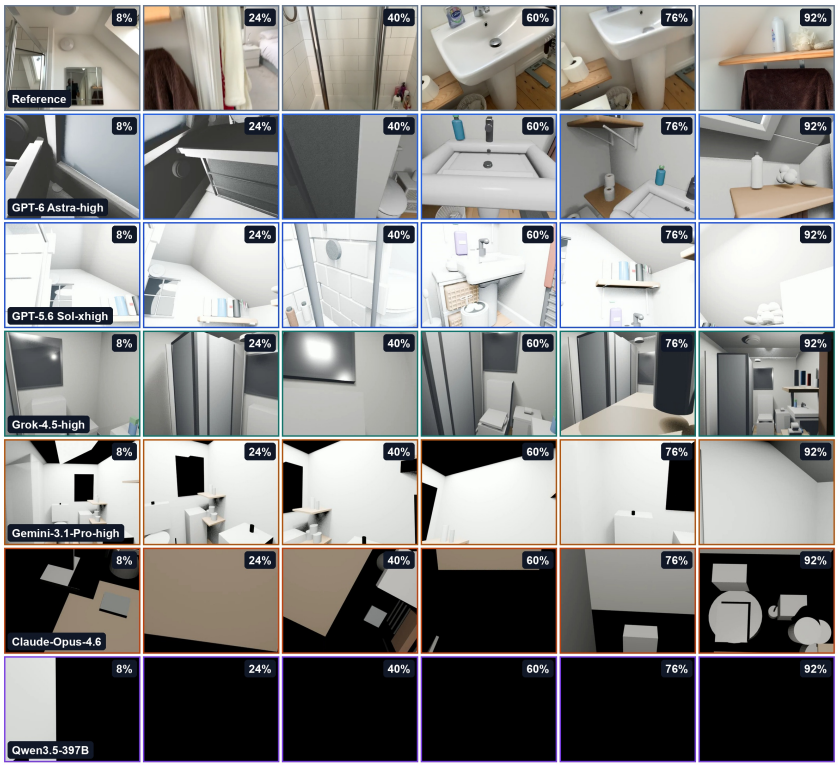


Figure 16: **Candidate 3: bathroom fixtures.** ARKitScenes scene 45261182 at six matched timestamps.

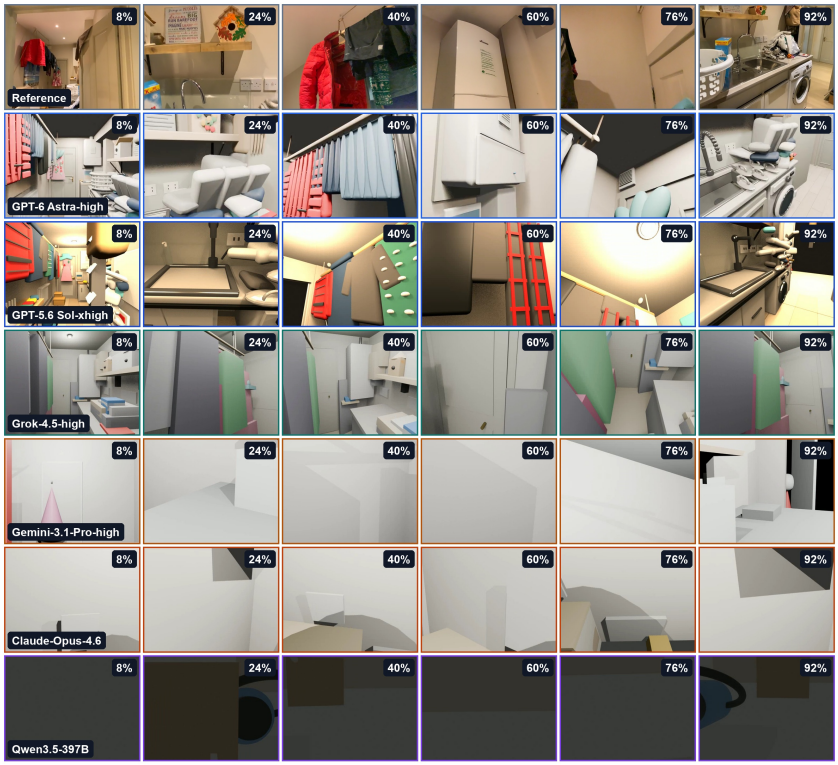


Figure 17: **Candidate 4: cluttered storage room.** ARKitScenes scene 42898817 at six matched timestamps.

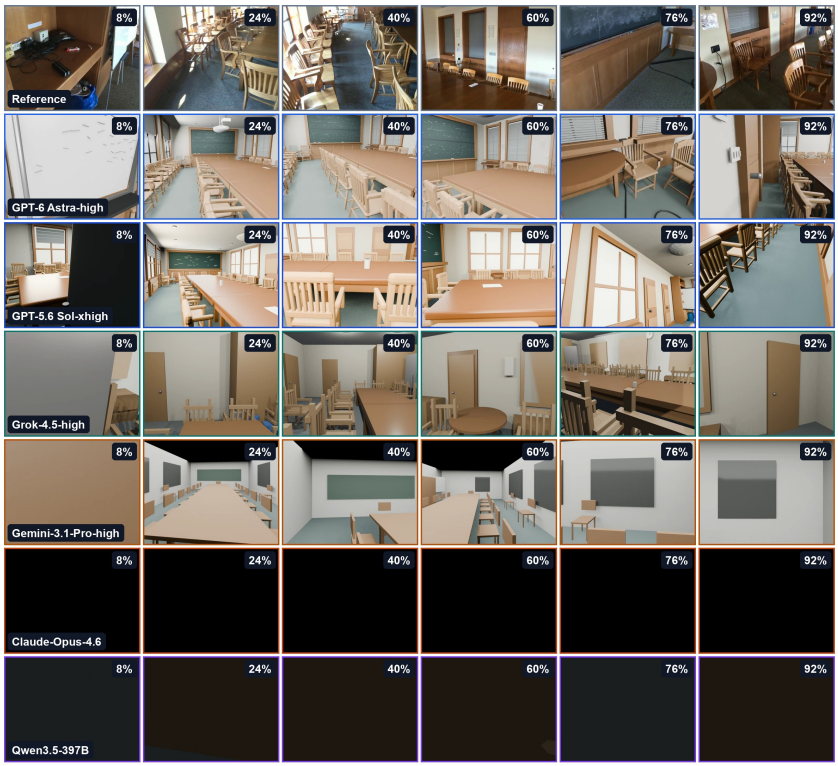


Figure 18: **Candidate 5: repeated-instance classroom.** ScanNet scene scene0500_00 at six matched timestamps.

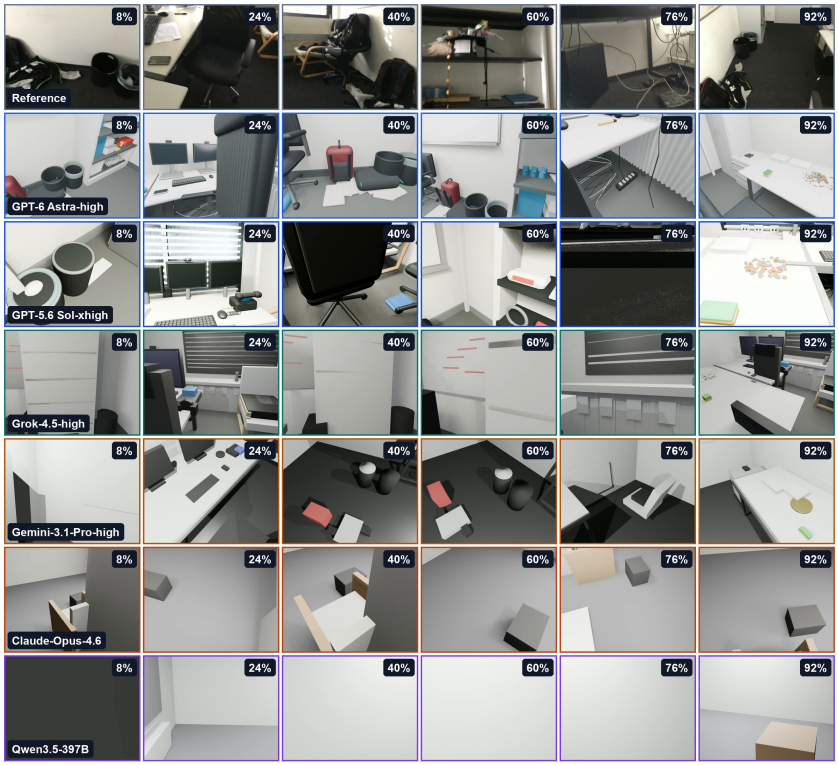


Figure 19: **Candidate 6: cluttered desktop.** ScanNet scene scene0700_02 at six matched timestamps.

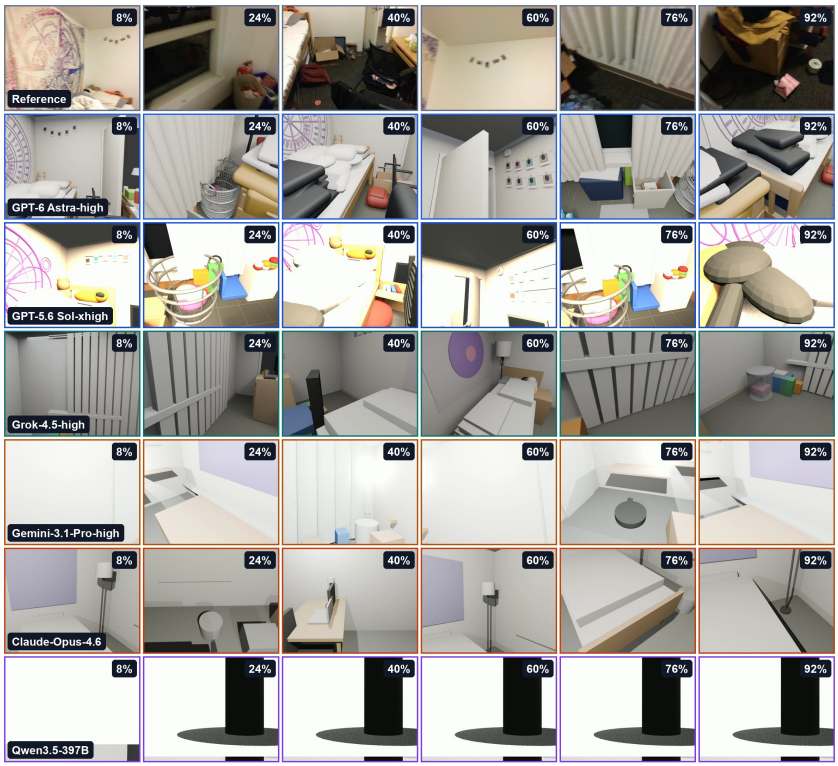


Figure 20: **Candidate 7: bedroom workspace.** ScanNet scene scene0695_00 at six matched timestamps.



Figure 21: **Candidate 8: bathroom counter.** ScanNet scene scene0664_02 at six matched timestamps.

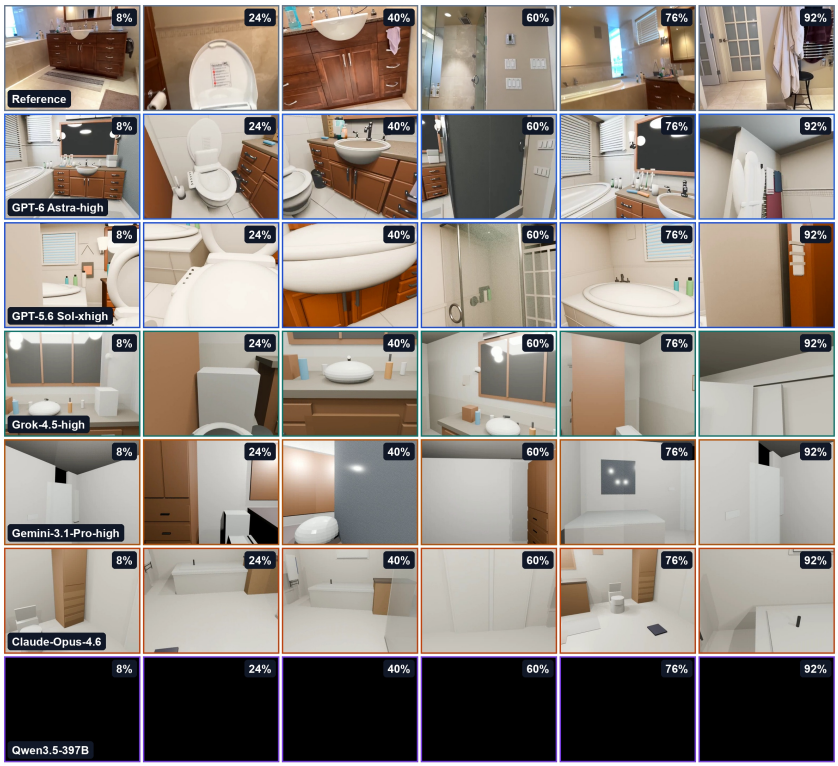


Figure 22: **Candidate 9: bathroom and doorway.** ScanNet++ scene e7af285f7d at six matched timestamps.

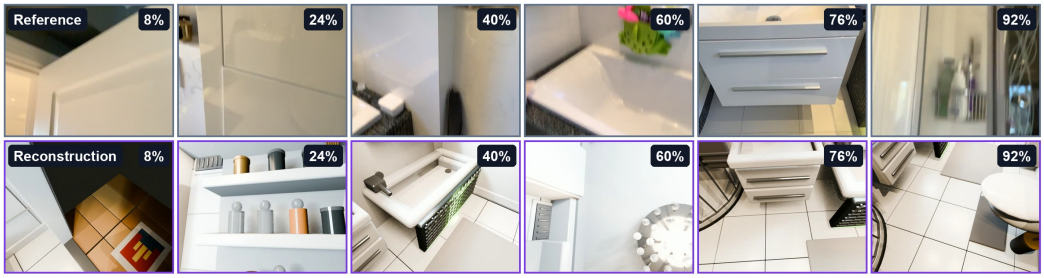


Figure 23: **Candidate 10: shared office.** ScanNet++ scene acd95847c5 at six matched timestamps.



Figure 24: **Candidate 11: meeting room.** ScanNet++ scene 5748ce6f01 at six matched timestamps.

Metric-complementarity cases. The single-configuration strips in Figures 25 and 26 provide focused source-versus-reconstruction views of four localized semantic failures. They illustrate metric complementarity and do not estimate failure prevalence.



(a) Scene 47429912: size and room-scale retention

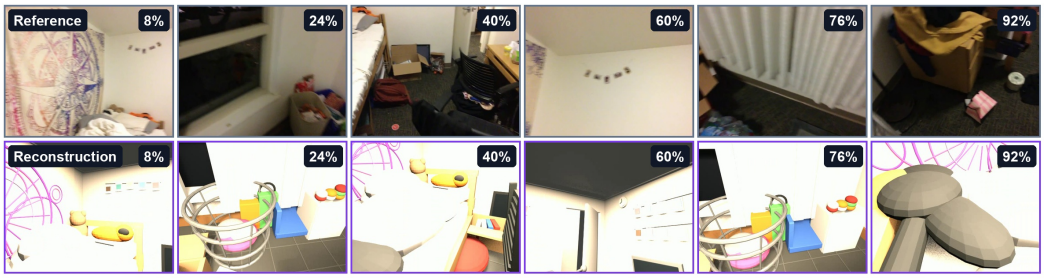


(b) Scene 42446049: direction and route retention

Figure 25: **High perceptual similarity does not prevent localized semantic loss.** Panel (a) reaches 89.4 LS but retains neither of its two source-correct size questions. Panel (b) reaches 89.0 LS but misses its source-correct room-size, relative-direction, and route questions.



(a) Scene scene0500_00: object-count retention



(b) Scene scene0695_00: appearance-order retention

Figure 26: **Inventory and temporal coverage remain distinct failure modes.** Panel (a) reaches 84.2 LS but misses its source-correct object-count question. Panel (b) reaches 81.8 LS but retains none of three source-correct appearance-order questions and misses its route question.