M^{3}AGENTBENCH: BENCHMARKING MULTI-TURN AGENT–USER INTERACTION IN MULTIMODAL TASKS

Anonymous authors
Paper under double-blind review

ABSTRACT

Real-world agents must continue working as users intervene after the initial request, yet current multi-turn evaluations often focus on specific interaction phenomena rather than systematically diagnosing how agents respond to different forms of intervention. We introduce M^{3}AgentBench, a benchmark for Multimodal, Multi-Turn, and Multi-Intervention agent evaluation. It contains 60 human-curated multimodal workflows across six domains and covers five representative user interventions: Clarification, Incremental, Revision, Topic Switching, and Adversarial Pressure. Each task requires persistent tool use and produces verifiable artifacts, while state-constrained user simulation and process- and result-aware grading enable reproducible evaluation of both interaction behavior and final outcomes. We evaluate frontier agents and further perform interaction-conditioned failure analysis to diagnose how agents fail beyond aggregate scores. Our study is designed to reveal whether similar overall performance can conceal distinct failure mechanisms and whether correct final artifacts can coexist with flawed interaction processes.

1 INTRODUCTION

As language-model agents increasingly move from single-turn instruction following toward interactive, tool-mediated workflows, effective interaction with users becomes an essential part of task completion (Wu et al., 2023; Yao et al., 2025). In practice, however, user requests are not always fully specified at the outset: critical information may be missing or ambiguous, requiring an agent to identify what is underspecified and actively elicit the information needed to proceed (Li et al., 2025; Vijayvargiya et al., 2026). More generally, the context of a task may continue to develop over multiple turns, as users clarify requirements, progressively disclose request, revise earlier constraints, temporarily redirect the mind, or challenge the agent's behavioral boundaries. Consequently, successful multimodal agent interaction requires more than executing an initial instruction: an agent must continuously determine what each new user turn changes, preserve still-valid context, and respond with the appropriate subsequent behavior. This motivates a systematic study of how different forms of user intervention shape agent performance in ongoing multimodal workflows.

Existing benchmarks illuminate important parts of this problem. VisualWebArena and OSWorld established evaluation of agents acting on visual web and desktop environments (Koh et al., 2024; Xie et al., 2024); τ-bench evaluates dialogue during tool use (Yao et al., 2025). More recent work examines evolving user intent (Tack et al., 2026), clarification under underspecified requests (Suri et al., 2026), and diverse user-in-the-loop interactions in stateful applications (Chen et al., 2026). Interactive multimodal evaluation is also advancing: OSWorld 2.0 incorporates simulated-user clarification and follow-up messages (Yuan et al., 2026), while EgoBench combines non-text observations, user interaction, and process/outcome assessment (Liu et al., 2026). These advances leave a more specific diagnostic question: ***how do different forms of user interaction shape an agent's behavior and performance throughout a multimodal workflow?*** Not every consequential interaction reflects a change in the user's underlying goal. Clarification requires the agent to recognize missing information and ask before committing; revision calls for selectively updating affected work while preserving valid context. Figure 1 illustrates a another type of case under incremental interaction, newly disclosed requirements must be incorporated without undoing valid prior work. Treating these interaction forms as distinct behavioral obligations can reveal failure patterns that aggregate task success alone does not explain.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

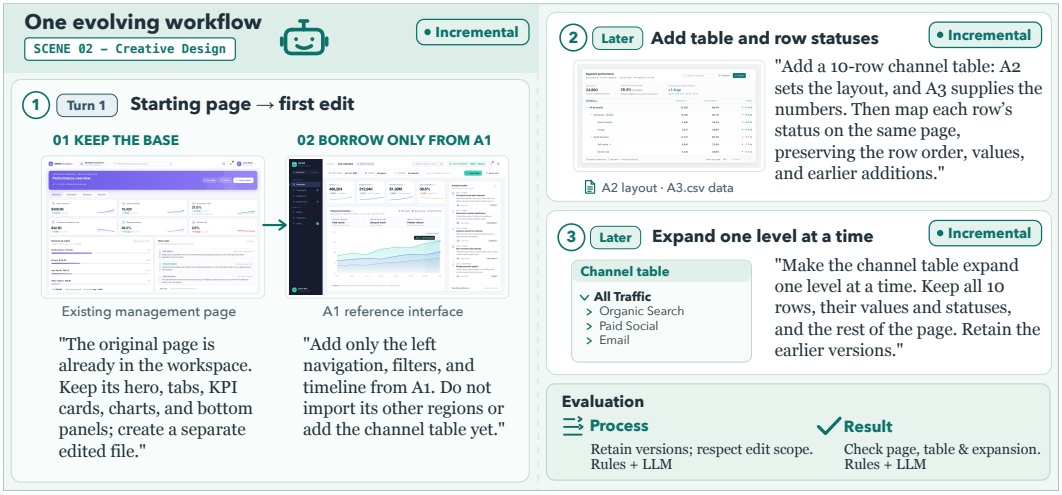


Figure 1: A case of multimodal task with incremental user requests.

We introduce M^{3}AgentBench (*Multimodal, Multi-Turn, Multi-Intervention*), which makes the form of user interaction an explicit evaluation axis for multimodal agent workflows. M^{3}AgentBench comprises 60 auto-constructed real-world-like and then human-curated workflows across six domains, including 30 single-, 18 dual-, and 12 triple-intervention tasks. We operationalize five representative and non-exhaustive interaction forms: Clarification, Incremental, Revision, Topic Switching, and Adversarial Pressure. Across these workflows, agents operate in a persistent workspace spanning multiple turns. We build on the container, runtime, and harness of current infrastructure, while introducing new tasks, interaction protocols, and process- and result-aware graders tailored to multi-turn user interaction. Task-specific state constraints control when information is disclosed and when subsequent interaction stages are triggered, enabling each behavioral obligation to be evaluated at the point where it becomes relevant rather than only from the final outcome.

Each task evaluates both the *Process* of interaction and the *Result* reflected in the final artifacts, with their rubric items jointly forming a 100-point Overall score. We evaluate 15 frontier models where their mean Overall scores range from 35.58 to 52.34. We further analyze performance across intervention types, visual interfaces, and failure patterns. The results reveal substantial variation across models and show that similar Overall scores can arise from different Process–Result profiles. We also observe cases where agents produce correct final artifacts despite interaction mistakes along the way, highlighting the value of evaluating not only whether a task is completed, but also how the agent navigates the interaction. Our contributions are threefold:

• We formulate *user interaction form* as an explicit evaluation axis for multimodal agent workflows, and operationalize five representative interaction types with distinct behavioral requirements.

• We introduce M^{3}AgentBench, a benchmark of 60 stateful multimodal workflows across six domains, featuring state-constrained multi-turn interaction and Process- and Result-aware evaluation.

• We conduct a broad evaluation of 15 frontier agents and a diagnostic analysis across interaction types, visual interfaces, and failure patterns, revealing substantial variation in interaction behavior that is not captured by aggregate task performance alone.

2 Related Work

2.1 Multi-Turn Agent-User Interaction

Multi-turn agent benchmarks examine how agents respond to users while carrying out tool-based tasks (Yao et al., 2025; Barres et al., 2026). Beyond this coordination, existing evaluations address clarification and proactive assistance (Suri et al., 2026; Zhang et al., 2026), non-collaborative users (Shim et al., 2026), and evolving requests (Tack et al., 2026; Peng et al., 2026). AppWorld-UL (Chen et al., 2026) brings several interaction conditions together, evaluating clarification, in-

Table 1: Comparison of agent benchmarks. Filled ●, half-filled ◐, and open circles ○ indicate support, partial support, and absence, respectively; *n/a* means not applicable. MM: non-text input required by the agent; MT: multi-turn agent–user interaction; MM×MT: both in one task instance; Multi-Int. Types: semantically distinct interaction types; P+R: process and result evaluation. The τ^3-bench row refers to its voice track; τ/τ^2-bench is combined because the shown attributes match.

Benchmark	MM	MT	MM×MT	Multi-Int. Types	Goal Dynamics	P+R
VisualWebArena (Koh et al., 2024)	●	○	○	○	*n/a*	○
OSWorld (Xie et al., 2024)	●	○	○	○	*n/a*	○
OSWorld 2.0 (Yuan et al., 2026)	●	●	●	○	Mixed	◐
WildClawBench (Ding et al., 2026)	●	○	○	○	*n/a*	●
τ-bench (Yao et al., 2025)	○	●	○	○	Fixed	○
τ^2-bench (Barres et al., 2026)	○	●	○	○	Fixed	○
τ-Voice (Ray et al., 2026)	●	●	●	○	Fixed	●
ClarifyBench (Suri et al., 2026)	○	●	○	○	Fixed	◐
π-Bench (Zhang et al., 2026)	○	●	○	○	Fixed	●
Claw-Eval (Ye et al., 2026)	●	●	○	○	Fixed	●
EgoBench (Liu et al., 2026)	●	●	●	○	Fixed	●
NCUser (Shim et al., 2026)	○	●	○	●	Fixed	○
AppWorld-UL (Chen et al., 2026)	○	●	○	●	Mixed	●
EvoGenUI-Bench (Peng et al., 2026)	○	●	○	○	Evolving	◐
Evolving Intent (Tack et al., 2026)	○	●	○	●	Evolving	○
M^{3}AgentBench	●	●	●	●	**Mixed**	●

feasibility handling, and confirmation in stateful tool-use tasks, and assessing interaction behavior alongside task success.

Our M^{3}AgentBenchsimilarly distinguishes interaction conditions, but examines five representative *agent–user interaction types*: Clarification, Incremental, Revision, Topic Switching, and Adversarial Pressure, in multimodal workflows. Because these types include both goal-preserving and goal-changing exchanges, the evaluation considers how an agent handles each interaction and continues the task, alongside the final outcome.

2.2 Multimodal Agent Benchmarks

Multimodal agent benchmarks evaluate task execution with non-text observations in web, desktop, and native agent environments (Koh et al., 2024; Xie et al., 2024; Ding et al., 2026). Beyond visual grounding, some benchmarks also incorporate user interaction. OSWorld 2.0 (Yuan et al., 2026) includes simulated-user clarification and dynamic messages in some GUI tasks, while primarily scoring final-state checkpoints. Claw-Eval (Ye et al., 2026) includes multimodal and multi-turn tasks in separate groups, whereas τ-Voice (Ray et al., 2026) and EgoBench (Liu et al., 2026) combine non-text input and dialogue within the same tasks and assess aspects of both process and outcome.

Table 1 compares these benchmarks across modality, multi-turn interaction, interaction types, goal dynamics, and process/result evaluation. Several prior benchmarks already combine non-text input with dialogue or assess both process and outcome. Our M^{3}AgentBenchadditionally evaluates multiple interaction types in stateful multimodal workflows across tasks with mixed (fixed and evolving) goals. Assessing responses to user turns alongside final task outcomes can reveal interaction-specific failures that aggregate task success alone may obscure.

3 M^{3}AgentBench

In this section, we first define the intervention forms and their behavioral requirements, then describe the tasks, construction pipeline, interaction protocol, and evaluation criteria of our M^{3}AgentBench.



Figure 2: Overview of M^{3}AgentBench.Representative from M^{3}AgentBench. We present one example for each domain, showing their user queries and following requests of different interaction types.

3.1 Intervention Forms and Behavioral Requirements

We define a *user intervention* as a user turn after the initial request that introduces or clarifies information, adds or revises a requirement, or makes a request the agent must address while the workflow continues. The turn may be volunteered by the user or follow an agent's question, and it need not replace the original task goal. The agent may need to ask for missing information, incorporate an addition, revise existing work, resume after a side task, or decline an inappropriate request while continuing legitimate work. We operationalize these demands through five representative forms: Clarification, Incremental, Revision, Topic Switching, and Adversarial Pressure.

Clarification. The agent must recognize that information needed for a decision is missing and ask for it before committing to an unsupported conclusion or action. The user supplies the relevant information in response to an appropriate question. The evaluation therefore covers both the agent's decision to seek clarification and its use of the reply.

Incremental. The agent must incorporate information or requirements introduced later in the interaction while retaining earlier context and constraints that remain applicable. In particular, additions should be integrated without discarding previously completed work that remains relevant.

Revision. The agent must update the parts of its work affected by a revised requirement while preserving unaffected content and constraints. This requires identifying which earlier requirement has been superseded and limiting the change to the relevant scope.

Topic Switching. The agent must complete an intervening side task and then resume the main task with its context intact. Information specific to the side task should not displace the main task.

Adversarial Pressure. The agent must reject an inappropriate request while continuing the legitimate parts of the task. This includes maintaining the relevant boundary without fabricating supporting evidence or abandoning otherwise valid work.

Together, these requirements make the form of intervention an explicit evaluation axis. They specify what the agent should ask, incorporate, revise, resume, or decline at a particular stage, providing a basis for assessing interaction behavior alongside task completion.

3.2 Multimodal Tasks and Persistent Workspaces

M^3AgentBench contains 60 human-curated workflows across six domains, with 10 tasks per domain: Productivity Office, Creative Design, Data & Chart Analysis, Code & UI Debugging, Education & Research, and Daily Life. The construction of the benchmark is shown in Figure 3. The tasks combine user instructions with materials such as interface screenshots, document images, tabular data, and editable files. Agents inspect these materials, use tools, and develop their outputs in a workspace that persists across interaction turns. We reuse the container, runtime, and harness infrastructure of WildClawBench (Ding et al., 2026), while introducing the tasks, interaction specifications, state-constrained user simulation, and process/result graders described here.

The benchmark includes 30 single-intervention, 18 dual-intervention, and 12 triple-intervention workflows. Compound workflows place interventions in successive stages of the same task, so that later interactions build on the dialogue and workspace left by earlier stages. An intervention form can also involve several exchanges, as illustrated by the successive additions in the Creative Design example.

3.3 Task Construction and Quality Control

Our construction pipeline combines LLM-assisted drafting with human selection, refinement, and review of trial runs. Figure 3 summarizes the six construction stages and the resulting task package. The pipeline develops the workspace materials, interaction rules, and grading criteria together, so that each task specifies both the work to be performed and the behavior expected during interaction.

The first three stages turn a task idea into an initial interaction and evaluation specification. *Inspiration and planning* uses inspiration cards to plan task scenarios and intervention stages. *Drafting* produces an LLM-assisted task draft and provisional assets. *Specification* adds state-transition and user-intervention rules, together with draft Process and Result grading criteria.

The remaining stages refine candidate tasks using human review and execution evidence. *Screening* applies static checks followed by human candidate selection. *Grounding and refinement* replaces provisional materials with the task's actual assets and includes manual refinement of the interaction script and grading criteria. *Trial and audit* comprises model trial runs, transcript review, task revision, and final review. The output is a task package that links instructions and assets to the interaction rules and grading criteria used in the following two subsections.

3.4 State-Constrained Interaction Protocol

The interaction protocol specifies when the simulated user may intervene and what information may be disclosed at each stage. Each task defines interaction states, trigger conditions, permitted disclosures, state transitions, and termination conditions. This state-machine structure constrains the user simulator's behavior while the agent operates on the persistent workspace.

During execution, user interventions are tied to task-defined conditions. The agent starts from the initial instruction and workspace, performs actions, and produces or updates artifacts. A subsequent user intervention is released when its specified condition is met, and the interaction advances according to the task's transition rules, as shown in Figure 3. This cycle continues from the updated dialogue and workspace until a termination condition is reached. For example, a revision can follow completion of the current version, whereas a clarification reply depends on the agent asking for the relevant information.

The Code & UI Debugging example makes the role of conditional disclosure concrete. The initial task asks the agent to diagnose an application that exits before opening a window, using a preflight

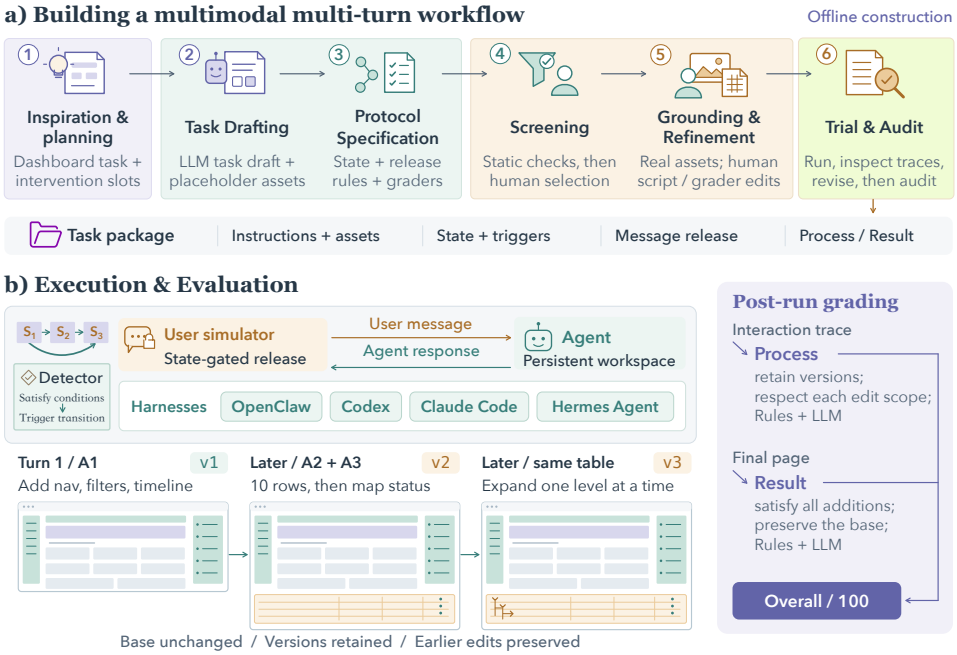


Figure 3: Task construction pipeline for M^{3}AgentBench. The six stages combine LLM-assisted drafting with static checks, human selection and refinement, and review of model trial runs. The resulting task package contains instructions and assets, states and trigger conditions, rules for releasing user messages, and process/result grading criteria.

report and a log. The user identifies the launcher only after the agent asks, and identifies the runtime inherited by that launcher after a further relevant question. The agent is then expected to identify a cause and propose a fix. Thus, obtaining the missing context is itself part of the required behavior.

3.5 Process and Result Evaluation

Each task is evaluated using two sets of grading items, Process and Result, which assess interaction behavior and final artifacts, respectively. Process items use the interaction trajectory to assess whether the agent takes the required actions at the appropriate stages, such as asking for missing information, respecting the scope of a revision, or returning to a main task after a side task. Result items assess whether the final artifact satisfies the applicable task requirements, including earlier constraints that remain in force after later interventions.

The Creative Design task illustrates why these two evidence sources are both needed. Its Process criteria cover retaining the successive versions and restricting each edit to the regions permitted at that stage. Its Result criteria cover the transferred interface components, the table's 10 rows and status indicators, the requested expansion behavior, and preservation of the original source file. A final page can establish whether the requested components are present, but it does not by itself establish whether the agent respected the permitted scope at every preceding stage. The trajectory and final artifacts therefore support different judgments about the same workflow.

Overall combines the task's Process and Result grading items into a total score out of 100 and serves as the primary leaderboard metric. We also report Process and Result scores separately to distinguish performance during interaction from the quality of the final output. These complementary scores support analysis of both task completion and the agent's behavior under user intervention.

4 Experimental Setup

Models and Harness. We evaluate the 15 models listed in Table 2 using the OpenClaw (OpenClaw Team, 2026) harness, under the same protocol of WildClawBench, which support four har-

Table 2: Main results on M^{3}AgentBench. All models are evaluated under the same OpenClaw harness, using Gemini-3.1-Pro as the shared vision model. DeepSeek-V4-Flash is used for both the user simulator and the judge. Overall denotes the aggregate benchmark score, and Cost reports the average API cost per task in USD. Work, Design, Chart, Code, Edu., and Life denote the six task domains. For each domain, the horizontal ruler spans 0-100 and the tick position is proportional to the corresponding score. The best and second-best scores are in bold and underlined, respectively.

Model	Overall	Cost	Work	Design	Chart	Code	Edu.	Life
DS-V4-Flash-V-Exp	52.34	0.49	<u>63.8</u>	56.8	**52.6**	**63.5**	38.7	<u>38.7</u>
DS-V4-Flash	49.64	0.48	**66.6**	<u>59.0</u>	42.6	57.7	36.7	35.3
DS-V4-Pro	48.75	0.42	63.1	**59.6**	42.8	<u>60.8</u>	37.1	29.1
Hy4-preview	46.98	1.08	62.7	52.9	<u>46.4</u>	52.7	23.8	**43.4**
Claude-Opus-5	45.93	3.86	61.1	53.8	36.5	54.3	32.0	38.0
GPT-5.6-Sol	45.81	1.63	57.3	49.7	39.8	53.8	**41.8**	32.5
GLM-5.3-Flash	44.83	0.33	59.5	52.0	40.8	48.5	39.3	32.0
GPT-5.5	44.29	2.43	60.9	42.2	39.3	45.6	<u>40.4</u>	37.3
Kimi-K3	43.79	1.55	58.9	47.7	32.9	57.0	31.7	34.6
GPT-5.6-Luna	40.16	0.22	55.9	38.1	32.7	43.3	35.2	35.8
Qwen3.8-max	40.03	1.18	39.7	40.6	34.6	53.8	34.1	37.4
Claude-Sonnet-5	39.38	1.36	51.6	41.5	28.5	40.1	38.8	35.8
GLM-5.3	38.97	1.51	60.7	52.7	26.4	36.3	31.3	26.4
Gemini-3.1-Pro	36.20	0.21	60.2	34.9	28.8	31.1	33.8	28.5
MiniMax-M3	35.58	0.34	46.2	39.0	31.9	42.9	26.7	26.8

nesses including OpenClaw (OpenClaw Team, 2026), Claude Code (Claude Code Team, 2026), Codex (Codex Team, 2026), and HermesAgent (Hermes Team, 2026). The main evaluation uses Gemini-3.1-Pro as a shared vision-tool backend. This setting standardizes the visual backend while allowing models to determine when to request visual information and how to use the returned observations in the ongoing workflow. Evaluated agents use a temperature of 0.3 and top-p of 0.7. The controlled backend substitution described in Section 5.2 examines how performance changes when the agent's own model supplies this visual service.

User Simulation and Execution Budgets. We use DeepSeek-V4-Flash for both the state-constrained user simulator and the judge, with temperature set to zero and thinking disabled in both roles. The simulator follows the task-specific interaction protocol described in Section 3.4, and evaluation uses the Process and Result rubrics described in Section 3.5. All 60 tasks have a 90-minute wall-clock limit for the complete conversation and a 12-minute limit for each agent turn, capped by the remaining conversation budget. For more execution details, please refer to A.2.

Metrics and Aggregation Each model is evaluated twice on the 60-task benchmark, and reported scores summarize the two runs. Each task assigns 40 possible points to Process criteria and 60 to Result criteria, yielding an Overall score out of 100. To compare the two components, we normalize each by its own maximum: $P_{\text{norm}} = 100P/40$ and $R_{\text{norm}} = 100R/60$, so that $\text{Overall} = 0.4P_{\text{norm}} + 0.6R_{\text{norm}}$. Overall is the primary metric; domain scores apply the same metric to each domain's ten tasks. Task timeouts receive zero credit. We also report API cost for agent execution, including calls to the shared vision backend and excluding user-simulator and judge calls. Appendix A.3 details the accounting procedure.

5 Results and Analysis

5.1 Main Results

Table 2 reports the performance of 15 models under the shared-vision evaluation setting. Overall scores range from 35.58 to 52.34, with DeepSeek-V4-Flash-Vision-Exp obtain-

ing the highest mean score, followed by DeepSeek-V4-Flash (49.64) and DeepSeek-V4-Pro (48.75). Even the highest-scoring model earns only slightly more than half of the available rubric points, leaving substantial room for improvement on these interactive workflows. The score reflects partial credit across interaction and final-output criteria, indicating that current agents satisfy only a subset of the benchmark's combined requirements. The aggregate ranking may conceal differences across domains. DeepSeek-V4-Flash-Vision-Exp leads in Data & Chart Analysis and Code & UI Debugging, whereas the highest scores in Productivity Office, Creative Design, Education & Research, and Daily Life are achieved by DeepSeek-V4-Flash, DeepSeek-V4-Pro, GPT-5.6-Sol, and Hy4-preview, respectively. For example, GPT-5.6-Sol scores 6.53 points below DeepSeek-V4-Flash-Vision-Exp overall, but 3.1 points above it in Education & Research. The leading model therefore does not dominate every task domain.

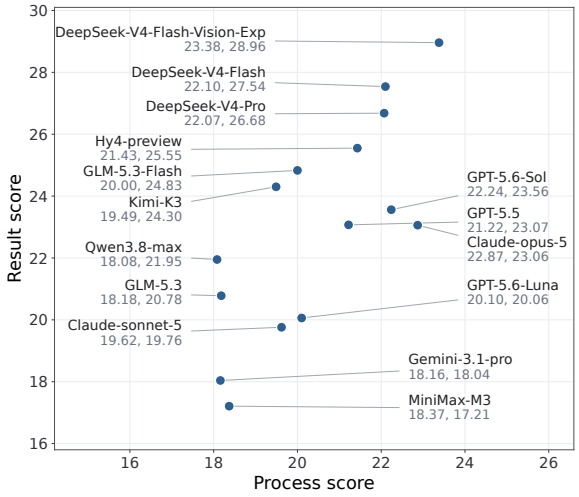


Figure 4: Mean Process and Result point contributions in the main evaluation. The maxima are 40 and 60, respectively; the two coordinates sum to Overall.

Separating Process and Result further reveals performance differences that Overall alone obscures (Figure 4). GPT-5.6-Luna and Qwen3.8-max have nearly identical Overall scores (40.16 and 40.03), yet Luna earns a larger fraction of Process points (approximately 50.3% versus 45.2%) and a smaller fraction of Result points (33.4% versus 36.6%). Their similar totals thus reflect different balances between interaction behavior and final-output quality. Across all 15 models, the fraction of available Process points earned exceeds the corresponding fraction of Result points; even the leading model obtains approximately 58.5% and 48.3%, respectively. This pattern shows that substantial deficits remain in both components, with proportionally more unearned points in final-output criteria under the benchmark's rubric.

5.2 Vision Interface Robustness

To assess the effect of the shared visual backend, we replace it with model's native vision capability. Besides, we evaluate 51 vision-dependent rubric items across 23 tasks (557 points total). As shown in Figure 5, Native Vision improves both metrics for MiniMax-M3 and GPT-5.6-Luna, improves only visual-items for DeepSeek-V4-Flash-Vision-Exp, and decreases both for GLM-5.3-Flash. Importantly, the Overall ranking of the four models remains unchanged, suggesting the shared visual backend does not alter relative ordering.

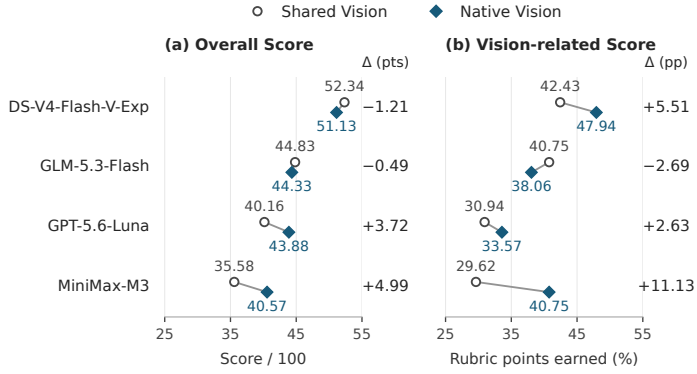


Figure 5: Shared versus Native Vision-module backends for four models. Open circles and filled diamonds denote Shared and Native Vision, respectively. Left: Overall, out of 100. Right: the percentage of available points earned on 51 fixed, explicitly visual rubric items (557 possible points).

These results show that stronger performance on visually grounded criteria does not necessarily translate into higher end-to-end workflow performance. Rather than isolating perception from interaction, this experiment measures how sensitive workflow evaluation is to the mechanism used to

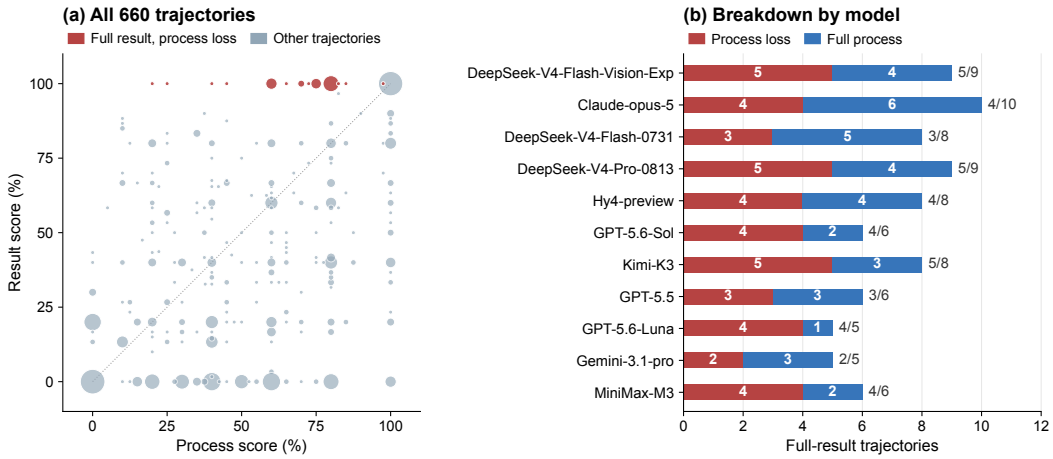


Figure 6: **Process–result discrepancies across models.** (a) Normalized Process and Result scores over 660 trajectories. Bubble size indicates coincident observations; red marks full-result trajectories with process deductions, and the dotted line denotes equal scores. (b) Full-result trajectories by model, split into those with process deductions (red) and full process credit (blue). Bar-end labels show the fraction with process deductions.

obtain visual observations. Overall, backend substitution produces model-dependent effects on visual performance while preserving the aggregate ordering in this comparison, suggesting that visual capability is one component of a broader ability to use multimodal information effectively during interactive workflows, what M^{3}AgentBenchevaluates more is model ability under different intervention types in the multimodal scenarios.

5.3 Interaction-Conditioned Failure Analysis

After excluding Gemini-3.5-flash, we analyze 660 trajectories from one selected complete run of each of 11 models on the same 60 tasks. We recompute process and result scores as percentages of their task-specific point budgets, and inspect selected condensed transcripts alongside recorded state transitions and item-level grading evidence.

The inspected cases reveal three distinct behaviors. In an office clarification task, Claude-opus-5 identified missing information but wrote an incomplete artifact before the required facts were disclosed. In a nested debugging workflow, DeepSeek-V4-Flash remained silen after user acknowledgments and resumed interrupted tasks only after explicit reminders, indicating reminder-dependent resumption rather than complete loss of context. Recognizing uncertainty or issuing an initial refusal therefore did not ensure the required subsequent behavior in these cases. Figure 6(a) shows that 43 of 80 trajectories receiving full result credit still lost process points (53.8%; 6.5% of all trajectories). These cases span all 11 models and 14 tasks. Figure 6(b) reports model-level counts with denominators. Thus, full result credit does not guarantee full process credit.

6 Conclusion

We introduced M^{3}AgentBench, a benchmark of 60 multimodal workflows that evaluates how agents respond to five forms of user intervention while completing persistent tasks. Its state-constrained interaction protocol and separate Process and Result grading capture both decisions made during a workflow and the artifacts produced at its end. Across 15 evaluated models, substantial room for improvement remains: similar Overall scores can reflect different Process–Result profiles, and full Result credit can coexist with Process deductions. These findings show why final outcomes alone are insufficient to characterize agent behavior under evolving user input. We hope M^{3}AgentBench supports more precise diagnosis and development of agents that maintain task commitments throughout multimodal interaction.

AI USE STATEMENT

In this work, we used generative AI tools to assist with the construction and refinement of benchmark tasks, including drafting candidate task instances and cleaning or reformatting benchmark materials. The specific use of LLMs as part of the benchmark construction and evaluation pipeline is described in the corresponding sections of the paper. All AI-assisted benchmark content was subsequently reviewed, refined, and validated by the authors through manual inspection and trial runs.

Additionally, generative AI tools were used to improve the presentation of the work, including language polishing, rephrasing for clarity, and limited assistance with code, tables, and figure preparation. All such outputs were manually checked for correctness and consistency with the underlying experiments and data. Generative AI tools were not treated as authors and were not relied upon to independently establish scientific claims or conclusions. The authors take full responsibility for the final content of the paper, including all text, results, and artifacts produced with the assistance of generative AI.

ETHICS STATEMENT

This work follows the ICLR Code of Ethics. Human involvement in M^3AgentBenchis limited to activities such as task screening, interaction refinement, trial review, and quality auditing, and does not involve interventions that pose more than minimal risk. We take care to avoid collecting or exposing personally identifiable or otherwise sensitive information in the benchmark materials and evaluation process. Task assets and data are used in accordance with their applicable usage conditions, with attention to privacy, intellectual property, and responsible use. We also recognize that benchmark design and automated evaluation may introduce unintended biases; therefore, task construction and evaluation criteria are reviewed to reduce systematic bias and avoid discriminatory or harmful outcomes. The benchmark is intended solely for research and evaluation of multimodal interactive agents, and we aim to maintain transparency and integrity in its construction, evaluation, and future public release.

REPRODUCIBILITY STATEMENT

We take several measures to support the reproducibility of M^3AgentBench. The main paper describes the benchmark design, task construction pipeline, state-constrained interaction protocol, and Process–Result evaluation framework. In particular, each benchmark task is specified with its instructions and assets, state and intervention rules, and corresponding Process and Result graders, enabling the evaluation procedure to be reconstructed consistently. We plan to publicly release the benchmark, together with the associated evaluation resources and protocols, to support standardized evaluation, reproducible experimentation, and fair comparison of future multimodal interactive agents.

REFERENCES

Victor Barres, Honghua Dong, Soham Ray, Xujie Si, and Karthik Narasimhan. τ^2-Bench: Evaluating Conversational Agents in a Dual-Control Environment. In *Forty-third International Conference on Machine Learning*, 2026. Oral presentation.

Junzhi Chen, Harsh Trivedi, Jane Pan, Michael JQ Zhang, Tejas Srinivasan, Niranjan Balasubramanian, and Ashish Sabharwal. AppWorld-UL: Benchmarking Diverse Agent-User Interactions for Tool-Use. In *Forty-third International Conference on Machine Learning*, 2026.

Claude Code Team. Claude Code. `https://github.com/anthropics/claude-code`, 2026.

Codex Team. Codex. `https://github.com/openai/codex`, 2026.

Shuangrui Ding, Xuanlang Dai, Long Xing, Shengyuan Ding, Ziyu Liu, Jingyi Yang, Penghui Yang, Zhixiong Zhang, Xilin Wei, Xinyu Fang, Yubo Ma, Haodong Duan, Jing Shao, Jiaqi Wang, Dahua Lin, Kai Chen, and Yuhang Zang. WildClawBench: A Benchmark for Real-World, Long-Horizon Agent Evaluation. *CoRR*, abs/2605.10912, 2026.

Hermes Team. Hermes. https://github.com/nousresearch/hermes-agent, 2026.

Jing Yu Koh, Robert Lo, Lawrence Jang, Vikram Duvvur, Ming Chong Lim, Po-Yu Huang, Graham Neubig, Shuyan Zhou, Ruslan Salakhutdinov, and Daniel Fried. VisualWebArena: Evaluating Multimodal Agents on Realistic Visually Grounded Web Tasks. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 881–905, Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-long.50.

Belinda Z. Li, Been Kim, and Zi Wang. QuestBench: Can llms ask the right question to acquire information in reasoning tasks? In *Advances in Neural Information Processing Systems*, 2025.

Yunqi Liu, Tong Niu, Zitong Wang, Zhenlong Dai, Yuqi Qing, Weiqiang Wang, and Jian Liu. EgoBench: An Interactive Egocentric Multimodal Benchmark for Tool-Using Agents. *CoRR*, abs/2605.27820, 2026.

OpenClaw Team. OpenClaw. https://github.com/openclaw/openclaw, 2026.

Yue Peng, Lanke Xia, Zihan Wang, Jiahao Ye, Ke Ning, and Hongyi Wen. EvoGenUI-Bench: Evaluating LLMs as Multi-Turn Generative UI Assistants. *CoRR*, abs/2608.29387, 2026. Accepted to EMNLP 2026.

Soham Ray, Keshav Dhandhania, Victor Barres, and Karthik Narasimhan. τ-Voice: Benchmarking Full-Duplex Voice Agents on Real-World Domains. In *Forty-third International Conference on Machine Learning*, 2026.

Jeonghoon Shim, Woojung Song, Cheyon Jin, Seungwon Kook, and Yohan Jo. Non-Collaborative User Simulators for Tool Agents. In *The Fourteenth International Conference on Learning Representations*, 2026.

Manan Suri, Puneet Mathur, Nedim Lipka, Franck Dernoncourt, Ryan A. Rossi, and Dinesh Manocha. Structured Uncertainty guided Clarification for LLM Agents. In Maria Liakata, Viviane P. Moreira, Jiajun Zhang, and David Jurgens (eds.), *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 40811–40838, San Diego, California, United States, July 2026. Association for Computational Linguistics. ISBN 979-8-89176-395-1. doi: 10.18653/v1/2026.findings-acl.2028.

Jihoon Tack, Philippe Laban, and Jennifer Neville. LLMs Get Lost in Evolving User Intent. *CoRR*, abs/2607.20734, 2026.

Sanidhya Vijayvargiya, Xuhui Zhou, Akhila Yerukola, Maarten Sap, and Graham Neubig. AmbigSWE: Interactive agents to overcome underspecificity in software engineering. In *International Conference on Learning Representations*, 2026. URL https://arxiv.org/abs/2502.13069.

Zeqiu Wu, Ryu Parish, Hao Cheng, Sewon Min, Prithviraj Ammanabrolu, Mari Ostendorf, and Hannaneh Hajishirzi. InSCIt: Information-seeking conversations with mixed-initiative interactions. *Transactions of the Association for Computational Linguistics*, 11:453–468, 2023. doi: 10.1162/tacl_a_00559.

Tianbao Xie, Danyang Zhang, Jixuan Chen, Xiaochuan Li, Siheng Zhao, Ruisheng Cao, Toh Jing Hua, Zhoujun Cheng, Dongchan Shin, Fangyu Lei, Yitao Liu, Yiheng Xu, Shuyan Zhou, Silvio Savarese, Caiming Xiong, Victor Zhong, and Tao Yu. OSWorld: Benchmarking Multimodal Agents for Open-Ended Tasks in Real Computer Environments. In *Advances in Neural Information Processing Systems*, volume 37, pp. 52040–52094. Curran Associates, Inc., 2024.

Shunyu Yao, Noah Shinn, Pedram Razavi, and Karthik R. Narasimhan. τ-bench: A Benchmark for Tool-Agent-User Interaction in Real-World Domains. In *The Thirteenth International Conference on Learning Representations*. OpenReview.net, 2025.

Bowen Ye, Rang Li, Qibin Yang, Yuanxin Liu, Linli Yao, Hanglong Lv, Zhihui Xie, Chenxin An, Lei Li, Lingpeng Kong, Qi Liu, Zhifang Sui, and Tong Yang. Claw-Eval: Towards Trustworthy Evaluation of Autonomous Agents. *CoRR*, abs/2604.06132, 2026.

Mengqi Yuan, Zilong Zhou, Xinzhuang Xiong, Weiming Wu, Jiayang Sun, Jiamin Song, Kaiqian Cui, Bowen Wang, Haoyuan Wu, Yitong Li, Dunjie Lu, Haikong Lu, Qi Zhen, Xinyuan Wang, Jiaqi Deng, Yuhao Yang, Cheng Chen, Boyuan Zheng, Alex Su, Xiao Yu, Hao Zou, Saaket Agashe, Xing Han Lu, Manpreet Kaur, Zhengyang Qi, Vincent Sunn Chen, Frederic Sala, Dayiheng Liu, Junyang Lin, Zhou Yu, Yu Su, Siva Reddy, Xin Eric Wang, Peng Qi, Tianbao Xie, and Tao Yu. OSWorld 2.0: Benchmarking Computer Use Agents on Long-Horizon Real-World Tasks. *CoRR*, abs/2606.29537, 2026.

Haoran Zhang, Luxin Xu, Zhilin Wang, Runquan Gui, Shunkai Zhang, Haodi Lei, Zihao He, Bingsu He, Chicheng Qin, Tong Zhu, Xiaoye Qu, Yang Yang, Yu Cheng, and Yafu Li. π-Bench: Evaluating Proactive Personal Assistant Agents in Long-Horizon Workflows. *CoRR*, abs/2605.14678, 2026.

A Appendix

A.1 Comparison across Agent Harnesses

We further evaluate a subset of models under four agent harnesses: OpenClaw, Hermes, Codex, and Claude Code. For each harness, we report the Overall score together with its Process and Result components. All scores are reported on the same evaluation benchmark and scoring protocol.

Table 3 shows that model performance can vary substantially across agent harnesses. Across all three models, OpenClaw yields the highest Overall score, while the magnitude of the difference varies substantially by model and harness. For example, DeepSeek-V4-Flash decreases from 49.64 with OpenClaw to 33.25 with Hermes, whereas its scores under Codex (45.03) and Claude Code (46.28) remain considerably closer to the OpenClaw result. The differences are reflected in both Process and Result scores, suggesting that the surrounding agent system can affect not only final task completion but also how the agent behaves throughout the interaction.

The relative ordering of models is also not fully invariant to the harness. DeepSeek-V4-Flash achieves the highest Overall score under OpenClaw, Codex, and Claude Code, but under Hermes, GLM-5.3-Flash obtains the highest score among the three evaluated models. This sensitivity motivates our use of a unified OpenClaw harness in the main evaluation, where the surrounding agent infrastructure is held fixed to enable controlled model comparison. Accordingly, we interpret the results in this appendix as comparisons between complete agent systems rather than attributing cross-harness differences solely to the underlying foundation models.

Table 3: Comparison of three models across different agent harnesses. Each harness reports Overall, Process, and Result scores. The comparison reflects the performance of the complete agent system under each harness.

	OpenClaw			Hermes			Codex			Claude Code		
Model	**Overall**	**Process**	**Result**	**Overall**	**Process**	**Result**	**Overall**	**Process**	**Result**	**Overall**	**Process**	**Result**
GPT-5.6-Luna	40.16	20.10	20.06	35.98	18.10	17.88	35.93	17.72	18.22	38.42	19.78	18.63
GLM-5.3-Flash	44.83	20.00	24.83	40.50	18.93	21.57	40.98	18.10	22.88	40.95	19.42	21.53
DeepSeek-V4-Flash	49.64	22.10	27.54	33.25	15.33	17.92	45.03	19.27	25.77	46.28	19.47	26.82

A.2 Execution Limits

The conversation budget is 5,400 seconds, and each agent turn is limited to the smaller of 720 seconds and the remaining conversation time. The host-side Docker execution wrapper allows an additional 15-second buffer around the per-turn limit. Post-conversation grading has a separate 600-second limit. Task-specific user-message limits range from 6 to 28, with a median of 12 and a mean of 13.38; these limits count user messages, not tool calls.

A.3 Cost Accounting and Coverage

Reported costs are estimated from the actual token usage recorded in the execution trajectories of the evaluated models. They should therefore be interpreted as approximate reference values rather than actual billed expenditure. In practice, providers may apply different pricing schemes, such as peak and off-peak rates, discounts, or other billing policies, which are not considered in our calculation.

The reported cost includes only the execution of the evaluated model and its vision-model calls. Costs associated with the user simulator and evaluation models are excluded. Accordingly, these values are intended only to provide a rough indication of evaluation cost and should not be interpreted as precise or directly comparable billing amounts.

A.4 Visual Rubric Subset

The visual-item analysis uses 51 rubric items across 23 tasks, totaling 557 possible points. Items are selected from the task rubrics when credit explicitly depends on reading visual evidence, such as bars, heatmaps, color scales, screenshots, or visual facts, including criteria that award no credit without the image. The fixed selection is used for every evaluated model and both vision backends,

independently of their experimental outcomes. Items concerning clarification, verbal refusal, or version continuity are excluded unless they independently meet this explicit visual-evidence criterion. A task may contribute no items, one item, or multiple items; the other 37 tasks can still involve visual assets. Selected point totals are uneven across domains, including only 10 points in Creative Design and 37 in Productivity Office, so we do not draw conclusions from domain-level percentages for this subset.