OMNIGUI: BENCHMARKING GUI AGENTS IN OMNI-MODAL SMARTPHONE ENVIRONMENTS

Anonymous authors
Paper under double-blind review

ABSTRACT

Current benchmarks for graphical user interface (GUI) agents predominantly rely on static screenshots. However, real-world smartphone interaction routinely requires agents to process transient audio cues and temporal video dynamics that are tightly coupled with the moment of action. To bridge this gap, we introduce OmniGUI, the first step-level benchmark designed to evaluate GUI agents in omni-modal smartphone environments. OmniGUI provides continuous, interleaved multimodal inputs—comprising static images, synchronous audio, and video clips—at every action step. The dataset encompasses 709 expert-demonstrated episodes (2,579 action steps) across 29 applications, systematically annotated with objective multimodal dependency levels. Because dedicated omni-modal GUI agent frameworks are currently in their nascent stage, we select foundational omni-modal models capable of natively processing interleaved inputs to serve as agent proxies for our initial baselines. Our empirical evaluation reveals that while current models exhibit competency on visually static tasks, their action prediction performance degrades significantly in environments requiring synchronous temporal and auditory signals. Furthermore, ablation studies isolate specific operational bottlenecks, notably cross-modal interference when processing task-irrelevant environmental noise. The complete dataset, evaluation pipeline, and baseline prompts are provided in the supplementary material for peer review. The full dataset will be publicly released upon publication.

1 INTRODUCTION

GUI agents—systems that perceive device interfaces and execute actions on behalf of users—have attracted growing research interest (Hong et al., 2024; You et al., 2024; Cheng et al., 2024). Powered by large foundational models, these agents interpret visual screens and perform operations such as tapping, swiping, and typing text, enabling task automation across smartphones (Rawles et al., 2023), desktops (Xie et al., 2024), and web browsers (Deng et al., 2023).

A number of benchmarks have been developed to evaluate GUI agent capabilities (Appendix D, Table 8). The majority of existing benchmarks provide only static screenshots as perceptual input. A few recent works have begun to incorporate additional modalities, introducing audio transcriptions (Zheng et al., 2024) or video recordings (Chen et al., 2024; Lin et al., 2024; Jang et al., 2024). Despite these advances, existing multimodal benchmarks largely treat audio and video as *pre-task reference content*—for example, watching an instructional video before task execution. However, real-world device interaction routinely involves multimodal signals that are tightly coupled with the moment of action. On a typical smartphone, users encounter transient notification sounds, specific video playback states, or voice assistant instructions that directly govern the subsequent operation. These step-specific temporal and auditory contexts cannot be fully captured by static screenshots or pre-recorded reference videos.

To address this gap, we introduce **OmniGUI** (Figure 1), the first benchmark designed to evaluate GUI agents receiving continuous, interleaved multimodal inputs—comprising static images, synchronous audio, and temporal video clips—at *every* action step in real-world smartphone environments. OmniGUI encompasses 709 expert-demonstrated episodes (comprising 2,579 action steps) across 29 mobile applications. To ensure structural validity, the dataset is formulated around five cognitive operational dimensions (e.g., Temporal Reasoning, Instant Response) and subsequently categorized

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.

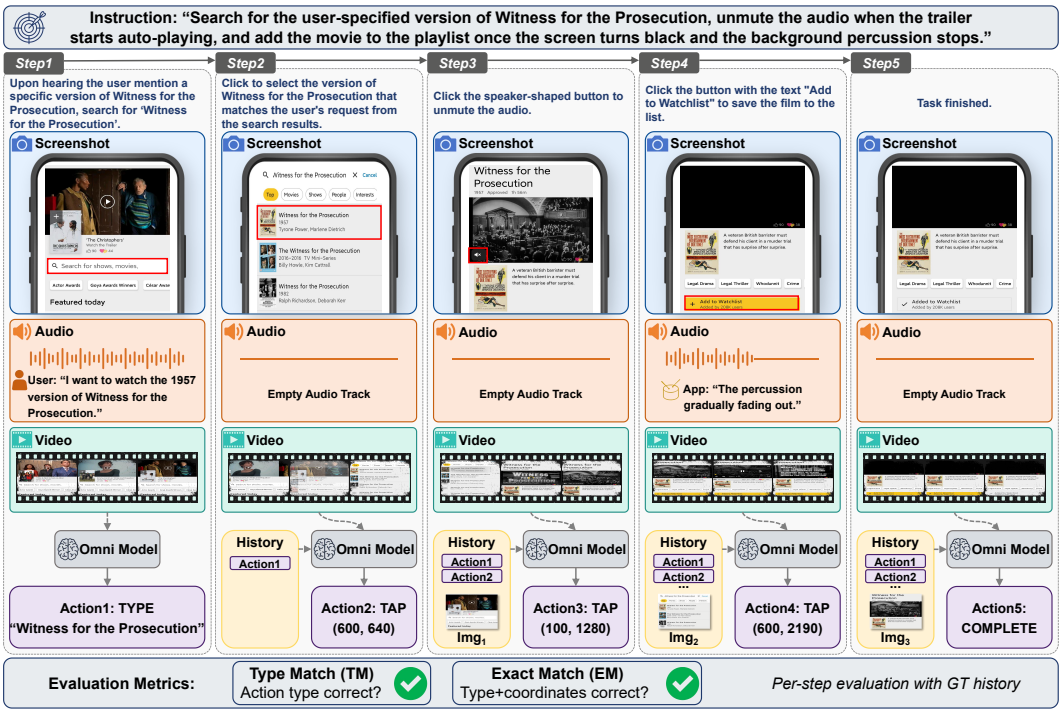


Figure 1: **Overview of the OmniGUI benchmark framework.** Given a multimodal instruction (e.g., a spoken request), a GUI agent interacts with a smartphone interface across multiple steps. At each action step, the agent receives interleaved inputs comprising a static screenshot, real-time audio (e.g., user speech, application sounds), and a temporal video clip, along with the action history. Based on these synchronous multimodal signals in the environment, the agent predicts the subsequent action (e.g., `TYPE`, `TAP`). Performance is quantitatively evaluated using Type Match (TM) and Exact Match (EM) metrics against ground-truth human demonstrations. Coordinates are screenshot pixels; `TYPE`/`COMPLETE` denote `INPUT`/`TASK_COMPLETE`.

into three objective multimodal dependency levels (AV-Critical, AV-Supportive, AV-Present) based strictly on physical information availability. At each step, the agent is required to predict a precise action primitive and its corresponding parameters (e.g., normalized coordinates, strings) from a comprehensive 13-action space.

Our primary objective is to evaluate how GUI agents operate within fully multimodal interactive environments. Since dedicated omni-modal GUI agent frameworks are currently in their nascent stages, we select foundational omni-modal models (e.g., Gemini 3.0 Pro, Qwen3-Omni) capable of natively processing interleaved inputs to serve as agent proxies for our initial baselines. Furthermore, in the absence of official GUI-specific reasoning protocols for these models, we implement a standardized, deterministic inference pipeline utilizing a unified prompt template. This design ensures evaluation fairness and rigorously isolates the step-level perception-to-action capabilities. By establishing this standardized protocol, OmniGUI provides a reproducible foundation for assessing future purpose-built omni-modal agent architectures.

Our extensive evaluation across eight proprietary and open-source models reveals critical insights into the current state of multimodal action execution. The highest-performing model achieves an Exact Match (EM) step accuracy of 63.6%, indicating that handling transient multimodal signals for precise step-level action prediction remains a significant challenge. Crucially, modality ablation studies empirically validate our dataset design: under No AV, EM decreases more on AV-Critical than on AV-Present tasks for all three ablated models, with two models improving on AV-Present tasks. Furthermore, the evaluation isolates specific operational bottlenecks in current architectures, such as cross-modal interference when presented with task-irrelevant multimodal signals, and significant performance degradation during concurrent dual-audio processing.

In summary, our contributions are as follows:

- We introduce OmniGUI, a GUI agent benchmark that provides interleaved image, audio, and video inputs at every action step, simulating the continuous multimodal perception required in real-world device interactions.
- We construct a high-quality, expert-demonstrated dataset of 709 episodes and 2,579 steps, systematically formulated around core HCI operational dimensions and rigorously annotated with objective multimodal dependency levels.
- We establish standardized initial baselines using foundational omni-modal models acting as agent proxies. Through comprehensive ablations, we validate the benchmark's structural necessity and identify specific operational bottlenecks (e.g., cross-modal interference) to provide empirical references for the development of future omni-agent frameworks.

2 Related Work

2.1 GUI Agent Benchmarks

The majority of existing GUI agent benchmarks rely exclusively on static screenshots as perceptual input. This includes extensive evaluations on Android (Rawles et al., 2023; Lu et al., 2025; Rawles et al., 2024), web browsers (Deng et al., 2023), desktop operating systems (Xie et al., 2024), and cross-platform element grounding (Cheng et al., 2024). While these works have established the foundation for agentic automation (Hong et al., 2024; You et al., 2024; Zhang et al., 2025), they fundamentally omit the auditory and temporal dynamics ubiquitous in real-world environments.

Recent efforts have begun incorporating non-visual modalities. Multimodal-Mind2Web (Zheng et al., 2024) augments web tasks with audio transcriptions, while GUI-World (Chen et al., 2024) and VideoGUI (Lin et al., 2024) introduce video demonstrations for interaction analysis. Most related to our work is VideoWebArena (Jang et al., 2024), which evaluates web agents using embedded multimedia content. However, these benchmarks predominantly treat audio and video as *pre-task reference materials* rather than step-level synchronous inputs. OmniGUI diverges fundamentally by targeting mobile environments where transient multimodal signals (e.g., sound alerts, video playback states) are tightly coupled with the exact moment of action, requiring continuous perception-to-action grounding at every step.

2.2 Omni-modal Foundation Models and Evaluations

The rapid evolution of foundational omni-modal models—capable of natively processing interleaved text, image, audio, and video—has been driven by both proprietary ecosystems (e.g., GPT-4o (Hurst et al., 2024), Gemini family (Team et al., 2024; Comanici et al., 2025; Gemini Team, 2025)) and open-source initiatives (e.g., Qwen3-Omni (Xu et al., 2025), MiniCPM-o (Yao et al., 2024), VITA (Fu et al., 2024)).

Consequently, numerous benchmarks have been proposed to evaluate their multimodal capabilities. These include comprehensive tri-modal understanding evaluations (Li et al., 2024; Wang et al., 2025), multimodal conflict diagnostics (Chowdhury et al., 2025), and broad audio-visual reasoning tasks (Fu et al., 2025a; Song et al., 2025; Yang et al., 2025; Sakshi et al., 2024). Despite rigorous evaluation across diverse domains, these benchmarks share a critical limitation: they strictly assess *passive perception and understanding*. The models output textual answers or classification labels based on fixed media inputs. None evaluate the sequential decision-making process where a model must translate dynamic, interleaved multimodal streams into executable operational primitives (e.g., coordinates, gestures) to alter the state of an interactive environment. OmniGUI bridges this exact gap, establishing a formal testbed for omni-modal agentic execution.

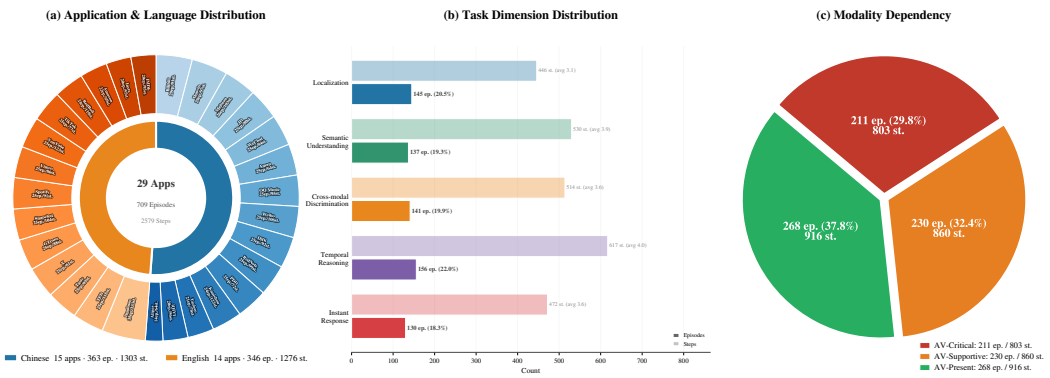


Figure 2: **Dataset statistics of OmniGUI.** (a) Application and language distribution, detailing the composition of 709 episodes and 2,579 fine-grained steps across 29 smartphone applications. (b) Distribution of episodes and steps across five core task dimensions, which are grounded in human-computer interaction and multimodal cognitive processes. (c) Episode distribution by multimodal dependency, with episode percentages and corresponding step counts.

3 The OmniGUI Benchmark

3.1 Interactive Environment and Formulation

We formulate the mobile GUI interaction as a sequential decision-making process. At each step t, the omni-modal agent receives a comprehensive observation state S_t from the environment and predicts an executable action a_t to fulfill a given natural language instruction G.

The observation state S_t is defined as a tuple of multimodal inputs: $S_t = (I_t, V_t, A_t, H_t)$, where:

- I_t is the high-resolution static screenshot captured at the current step t.
- V_t is the temporal video clip recording the screen dynamics from the previous action execution up to step t.
- A_t is the synchronous audio stream corresponding to V_t, capturing system sounds, media playback, or user voice commands.
- $H_t = \{a_1, a_2, \ldots, a_{t-1}\}$ represents the historical action trajectory.

Based on the instruction G and the multimodal state S_t, the agent generates an action $a_t \in \mathcal{A}$. As detailed in Table 7, the action space $\mathcal{A}$ encompasses 13 operational primitives across five categories: wait/observe (`NONE`), positional actions (e.g., `TAP`), gestural actions (e.g., `SWIPE_UP`), text input (`INPUT`), and system/status signals (e.g., `HOME`, `TASK_COMPLETE`). Continuous coordinate parameters (x, y) are normalized to a resolution-independent $[0, 1000] \times [0, 1000]$ scale.

3.2 Task Taxonomy and Dataset Statistics

The OmniGUI benchmark comprises 709 multi-step episodes, yielding a total of 2,579 fine-grained action steps (averaging 3.64 steps per episode). Constructed across 29 widely used smartphone applications, the dataset maintains a balanced bilingual distribution to assess cross-lingual generalization, including 15 Chinese applications (363 episodes, 1,303 steps) and 14 English applications (346 episodes, 1,276 steps), as illustrated in Figure 2(a) and detailed in Appendix B (Table 5). We organize the benchmark along two primary analytical axes: *task dimension* and *multimodal dependency*.

Task Dimensions and Formulation. To systematically evaluate the capabilities of omni-modal GUI agents, we established a top-down task taxonomy drawing upon Human-Computer Interaction (HCI) principles. We defined five operational dimensions that map the cognitive processing flow required for agentic execution—spanning perception, comprehension, reasoning, and reaction:

- **Localization (20.5% ep. / 446 steps):** Grounding actions to specific spatial coordinates based on visual or auditory descriptions.

- **Semantic Understanding (19.3% ep. / 530 steps):** Comprehending textual, visual, or spoken semantics to formulate multi-step execution plans.
- **Cross-modal Discrimination (19.9% ep. / 514 steps):** Synthesizing and aligning complementary information across video, audio, and text modalities.
- **Temporal Reasoning (22.0% ep. / 617 steps):** Tracking dynamic UI changes, moving elements, or event sequences over time.
- **Instant Response (18.3% ep. / 472 steps):** Reacting promptly to transient auditory or visual cues, such as alarms or specific video frames.

Guided by these five predefined dimensions, our annotators formulated the 709 goal-oriented episodes across 29 applications. This top-down formulation ensures that the collected tasks are not only ecologically authentic but also provide balanced coverage across different cognitive complexities.

Multimodal Dependency Taxonomy. To systematically quantify how omni-modal agents utilize non-visual signals, we categorize all episodes into three dependency levels (Figure 2c). This categorization is based solely on the objective information structure of the GUI environment (i.e., the physical availability of task-relevant signals) and is independent of empirical model performance. We define the following annotation codebook:

- **AV-Critical (29.8% ep. / 803 steps):** The correct action for at least one step cannot be determined from the static screenshot alone. The decision-critical information is exclusively present in the audio stream (e.g., a spoken instruction, a specific ringtone) or the temporal video stream (e.g., timing an action to a specific playback state).
- **AV-Supportive (32.4% ep. / 860 steps):** The static screenshot contains sufficient information to deduce the next action, but audio or video provides corroborating context that reduces ambiguity (e.g., background audio confirming an active media state). Non-visual signals improve robustness but are not strictly mandatory.
- **AV-Present (37.8% ep. / 916 steps):** Purely static UI tasks where all steps are fully resolvable from the static screenshot and action history. Audio and video modalities are present as environmental background noise and carry no additional task-relevant information.

Annotation Procedure and Quality Assurance. Following the task collection, we conducted a post-hoc evaluation to assign the multimodal dependency labels to each episode. To implement this, we established a strict modality-ablated annotation procedure. For each step, annotators were initially provided with only the static screenshot to determine if the correct action was unambiguously resolvable. Subsequently, the temporal video and audio streams were revealed, allowing them to finalize the objective dependency level based on whether the non-visual modalities introduced essential information.

To quantify the reliability of this taxonomy, a random subset of 100 episodes was independently annotated by a second reviewer. The process yielded a high inter-annotator agreement (Cohen's $\kappa = 0.84$), confirming substantial objective consensus. Disagreements in edge cases were resolved by a third senior annotator via majority vote.

3.3 Data Collection and Annotation Pipeline

The construction of OmniGUI follows a systematic pipeline designed to elicit diverse and high-quality human demonstrations.

Ten experienced Android users formulated tasks across 29 applications, recorded expert demonstrations, and annotated actions and modality dependencies. The synchronized logging and annotation procedures are detailed in Appendix B.

3.4 Evaluation Protocol and Metrics

We evaluate the models using a step-level teacher-forcing protocol, which isolates per-step multimodal perception capabilities from cascading compounding errors typical in autonomous rollouts. At each step t, the model receives the ground-truth history H_t and predicts a_t. Because our dataset

is built upon expert human demonstrations, achieving 100% performance conceptually represents perfect alignment with expert human operational intent. We employ four quantitative metrics:

- **Type Match (TM) [Step-level]:** Calculates the accuracy of predicting the correct action primitive (e.g., selecting `TAP` instead of `SWIPE_UP`), disregarding the specific parameters.
- **Exact Match (EM) [Step-level]:** A step is considered an exact match if both the action primitive and its associated parameters are correct. For positional actions, the predicted coordinates (x, y) must fall within the bounding box of the ground-truth target UI element. For text inputs, the generated string must exactly match the target text.
- **Success Rate (SR)[Episode-level]:** An episode is marked successful (1.0) if and only if the EM condition is satisfied for every single step within the trajectory; otherwise, it is 0.0.
- **Goal Progress (GP) [Episode-level]:** Measures the partial completion rate of a multi-step episode. It is calculated as the ratio of correctly executed steps (EM) to the total number of steps within that specific episode's ground-truth trajectory. This provides a granular, step-aware assessment for complex tasks even when the overall episode ultimately fails.

4 Experiments

We evaluate OmniGUI to assess its multimodal design and establish initial agent baselines. With dedicated omni-agent frameworks still emerging, foundational omni-modal models serve as agent proxies. We describe the experimental setup (Section 4.1), overall results (Section 4.2), modality ablations assessing the task taxonomy (Section 4.3), and qualitative errors (Section 4.4).

4.1 Experimental Setup

Evaluated Models. We evaluate state-of-the-art proprietary models: Gemini 3.0 Pro (Gemini Team, 2025), Gemini 3.0 Flash (Gemini Team, 2025), Gemini 2.5 Pro (Comanici et al., 2025), and Gemini 2.5 Flash (Comanici et al., 2025). We also evaluate leading open-source models: Qwen3-Omni (Xu et al., 2025), MiniCPM-o 4.5 (Yao et al., 2024), VITA-1.5 (Fu et al., 2025b), and Baichuan-Omni-1.5 (Li et al., 2025).[1]

Prompt Design and Input Structure. We use a unified Android GUI-agent prompt with interleaved multimodal context and ground-truth action history. Appendix A provides the prompt design and templates (Appendix A.1); Appendix B.1 gives annotated data examples.

Implementation Details. We use deterministic decoding where supported and model-specific API payload formatting. Generation settings are in Appendix A.2.

4.2 Main Results

Table 1 reports overall results and the complete dimension breakdown for all evaluated models.

Among the proprietary models, Gemini 3.0 Pro achieves the highest overall performance, yielding an Exact Match (EM) of 63.6% and a Success Rate (SR) of 33.4%. Despite being the state-of-the-art, its absolute success rate remains low, indicating that executing multi-step GUI tasks with interleaved transient multimodal signals remains a significant bottleneck for current models. Gemini 3.0 Flash follows closely, exceeding Pro in Temporal Reasoning TM, SR, and GP, but scoring lower in EM.

The evaluation reveals a substantial capability gap between proprietary and open-source models. Qwen3-Omni leads the open-source category with an EM of 32.3% and an SR of 5.1%, while the remaining open-source models struggle to complete full episodes successfully (SR $\leq$ 1.1%).

Across the five cognitive dimensions, performance varies consistently. Models generally exhibit higher Exact Match scores on static Localization tasks (e.g., 79.9% for Gemini 3.0 Pro) compared to Cross-modal Discrimination (59.9%) or Temporal Reasoning (61.8%) tasks. This variation is

[1] GPT-4o is excluded from the current evaluation. Its Chat Completions API lacks native support for interleaved raw audio-visual ingestion, while the Realtime API operates as a low-latency speech-to-speech stream, which is incompatible with the deterministic, step-level multimodal batch evaluation required by our benchmark protocol.

Table 1: **Comprehensive evaluation results on OmniGUI.** We report performance across four metrics: Type Match (TM), Exact Match (EM), Success Rate (SR), and Goal Progress (GP). The table presents both the **Overall** performance and a fine-grained breakdown across five specific task dimensions: Localization (*Local.*), Semantic Understanding (*Semantic Understand.*), Cross-modal Discrimination (*Cross-modal Discr.*), Temporal Reasoning (*Temporal Reason.*), and Instant Response (*Instant Resp.*). All metrics are reported as percentages (%). **Bold** indicates the best performance in each respective column. Model references are given in Section 4.1.

Model	Overall				Localization				Semantic Understand.				Cross-modal Discr.				Temporal Reason.				Instant Resp.			
	TM	EM	SR	GP	TM	EM	SR	GP	TM	EM	SR	GP	TM	EM	SR	GP	TM	EM	SR	GP	TM	EM	SR	GP
Proprietary Models																								
Gemini 3 Pro	**80.0**	**63.6**	**33.4**	**43.6**	**86.3**	**79.9**	**55.9**	62.6	**77.4**	**67.0**	**31.4**	**42.0**	**76.6**	**59.9**	**30.1**	**41.3**	78.9	**61.8**	22.7	36.9	**81.8**	**63.3**	**27.6**	**35.6**
Gemini 3 Flash	78.3	61.3	30.3	43.5	85.0	79.7	53.1	**63.1**	75.3	63.4	25.5	41.1	72.8	56.6	23.5	38.7	**80.0**	60.2	**25.3**	**39.4**	79.2	58.7	22.8	34.2
Gemini 2.5 Pro	75.7	44.1	15.5	26.3	86.1	62.7	31.7	41.5	72.8	42.9	11.7	22.4	70.6	41.7	13.2	25.1	73.8	45.9	9.7	22.5	76.6	42.8	11.0	19.5
Gemini 2.5 Flash	69.5	37.8	12.4	24.5	75.1	57.0	29.0	42.6	70.4	36.9	8.0	18.2	64.9	38.0	11.8	25.3	67.7	36.2	9.1	21.8	71.0	35.0	3.9	13.7
Open-source Models																								
Qwen3-Omni	63.1	32.3	5.1	17.4	65.7	43.9	10.3	28.5	58.3	32.1	2.9	14.0	57.9	24.6	2.2	13.2	66.2	32.4	5.8	16.8	67.4	33.3	3.9	13.7
VITA-1.5	39.3	12.1	1.1	2.2	48.4	16.5	2.8	3.9	43.4	19.0	2.2	3.2	33.9	13.4	0.0	0.8	35.4	8.1	0.6	2.0	36.9	11.8	0.0	0.8
MiniCPM-o-4.5	32.8	4.8	0.1	1.4	34.8	7.7	0.7	2.2	34.7	4.7	0.0	1.0	25.2	3.9	0.0	2.2	34.8	4.4	0.0	0.6	33.3	3.3	0.0	0.8
Baichuan-Omni-1.5	17.0	3.3	0.0	0.4	19.5	4.7	0.0	1.0	16.2	4.0	0.0	0.5	12.9	1.0	0.0	0.0	18.2	2.7	0.0	0.2	18.2	4.0	0.0	0.5

consistent with the increased complexity of integrating dynamic temporal and auditory cues into precise spatial actions compared to traditional screenshot-only visual grounding. Appendix C adds capability profiles (Appendix C.1) and application-level results (Appendix C.2).

4.3 Ablation Analysis

To empirically validate our task taxonomy and observe how models utilize different modalities, we conduct two sets of ablation studies using representative proprietary (Gemini 3 Pro, Gemini 2.5 Flash) and open-source (Qwen3-Omni) models.

Modality Ablation Analysis. Table 2 presents the results of systematically masking audio and video inputs. The No AV results support the dependency taxonomy: EM losses are larger on AV-Critical than AV-Present tasks for all three evaluated models. For instance, completely removing audio and video inputs (*No AV*) causes substantial performance drops on AV-Critical tasks across all models (e.g., a 13.8 percentage-point Exact Match drop for Gemini 3 Pro). Conversely, for Gemini 3 Pro on purely static AV-Present tasks, removing these modalities yields a smaller EM decrease of 1.2 percentage points.

Furthermore, the ablation results expose a cross-modal interference phenomenon. For Gemini 2.5 Flash and Qwen3-Omni, providing the full multimodal input (I+A+V) on AV-Present tasks results in lower EM compared to providing the static image alone (*No AV*). Specifically, Gemini 2.5 Flash's EM score decreases from 48.3% to 40.8% (7.5 percentage points) when environmental audio and video are introduced. This empirically indicates that the inclusion of task-irrelevant multimodal signals can negatively impact action prediction accuracy in visually sufficient contexts.

Instruction Modality (Text vs. TTS). In realistic agent deployments, users often initiate tasks via spoken commands. Table 3 compares model performance when text instructions are replaced with Text-to-Speech (TTS) synthesized audio, while keeping all environmental multimodal inputs intact.

Both evaluated models show smaller TTS-induced EM decreases on AV-Present than on AV-Critical or AV-Supportive tasks. For Gemini 3 Pro, the AV-Present and AV-Critical decreases are 2.8 and 10.6 percentage points; Qwen3-Omni has its largest decrease on AV-Supportive tasks (7.2 points). This pattern suggests difficulty in concurrent multimodal processing: both models incur smaller EM losses when grounding a spoken instruction to a static image, and larger losses when processing it alongside task-relevant environmental audio or video.

Table 2: **Modality ablation across dependency levels.** Header percentages are episode shares. We report all metrics (%) and show the performance gap below each score (Δ = Ablated − Full). To highlight the most impactful modalities, for each model and metric (column), the most severe performance drop among the three ablation settings is marked in **bold red**. Conversely, the most significant anomalous performance gain is marked in **bold teal**, revealing strong cross-modal interference when irrelevant modalities are provided in Present tasks.

Model	Modality Input	AV-Critical (29.8%)				AV-Supportive (32.4%)				AV-Present (37.8%)				Overall (100%)			
		TM	EM	SR	GP	TM	EM	SR	GP	TM	EM	SR	GP	TM	EM	SR	GP
Proprietary Models																	
Gemini 3 Pro	Full (I+A+V)	76.9	62.7	33.2	42.2	79.6	67.4	34.4	45.0	84.7	70.1	33.0	44.4	80.0	63.6	33.4	43.6
	No Audio (I+V)	74.4 (-2.5)	55.9 (-6.8)	28.7 (-4.5)	39.6 (-2.6)	79.6	64.1 (-3.3)	34.1 (-0.3)	44.5 (-0.5)	85.0 **(+0.3)**	70.9 **(+0.8)**	36.2 (+3.2)	46.9 (+2.5)	79.2 (-0.8)	63.0 (-0.6)	32.7 (-0.7)	43.3 (-0.3)
	No Video (I+A)	69.1 (-7.8)	50.0 (-12.7)	16.0 **(-17.2)**	34.5 **(-7.7)**	74.6 **(-5.0)**	59.2 (-8.2)	25.2 **(-9.2)**	41.5 (-3.5)	81.3 **(-3.4)**	67.7 **(-2.4)**	35.1 (+2.1)	48.0 (+3.6)	74.4 (-5.6)	58.1 (-5.5)	24.4 **(-9.0)**	40.7 **(-2.9)**
	No AV (Img Only)	67.3 **(-9.6)**	48.9 **(-13.8)**	17.2 (-16.0)	34.5 **(-7.7)**	75.3 (-4.3)	59.0 **(-8.4)**	26.3 (-8.1)	41.0 **(-4.0)**	81.8 (-2.9)	68.9 (-1.2)	37.8 **(+4.8)**	48.8 **(+4.4)**	74.2 **(-5.8)**	58.0 **(-5.6)**	26.1 (-7.3)	40.8 (-2.8)
Gem. 2.5 Flash	Full (I+A+V)	66.1	39.7	13.9	26.3	70.0	41.6	14.1	25.9	73.9	40.8	8.6	20.5	69.5	37.8	12.4	24.5
	No Audio (I+V)	61.6 (-4.5)	32.7 (-7.0)	9.0 (-4.9)	22.1 (-4.2)	68.8 (-1.2)	36.7 (-4.9)	11.5 (-2.6)	23.6 (-2.3)	76.0 **(+2.1)**	41.7 (+0.9)	10.3 (+1.7)	23.2 (+2.7)	68.2 (-1.3)	36.8 (-1.0)	10.3 (-2.1)	23.0 (-1.5)
	No Video (I+A)	60.7 (-5.4)	26.4 **(-13.3)**	2.0 **(-11.9)**	14.9 **(-11.4)**	66.1 (-3.9)	31.9 **(-9.7)**	5.9 **(-8.2)**	17.5 **(-8.4)**	69.7 **(-4.2)**	37.2 **(-3.6)**	7.6 **(-1.0)**	20.3 **(-0.2)**	65.1 (-4.4)	31.4 **(-6.4)**	4.9 **(-7.5)**	17.3 **(-7.2)**
	No AV (Img Only)	57.2 **(-8.9)**	28.2 (-11.5)	4.1 (-9.8)	18.1 (-8.2)	65.4 **(-4.6)**	36.2 (-5.4)	8.5 (-5.6)	23.6 (-2.3)	72.1 (-1.8)	48.3 **(+7.5)**	15.7 **(+7.1)**	34.4 **(+13.9)**	64.2 **(-5.3)**	36.5 (-1.3)	8.7 (-3.7)	24.4 (-0.1)
Open-source Model																	
Qwen3-Omni	Full (I+A+V)	58.0	31.6	7.0	18.5	65.1	34.4	4.8	17.4	66.9	34.3	3.2	16.0	63.1	32.3	5.1	17.4
	No Audio (I+V)	57.9 (-0.1)	26.7 (-4.9)	5.7 (-1.3)	17.3 (-1.2)	64.0 (-1.1)	32.7 (-1.7)	4.4 **(-0.4)**	17.9 **(+0.5)**	67.2 **(+0.3)**	34.9 (+0.6)	3.2 (+0.0)	16.7 (+0.7)	62.7 (-0.4)	31.1 **(-1.2)**	4.5 **(-0.6)**	17.3 **(-0.1)**
	No Video (I+A)	57.1 (-0.9)	28.8 (-2.8)	4.5 (-2.5)	17.8 (-0.7)	62.6 (-2.5)	31.2 **(-3.2)**	5.6 **(+0.8)**	16.3 **(-1.1)**	66.0 **(-0.9)**	39.3 **(+5.0)**	7.0 **(+3.8)**	20.1 (+4.1)	61.6 (-1.5)	32.6 **(+0.3)**	5.6 **(+0.5)**	17.9 **(+0.5)**
	No AV (Img Only)	54.7 **(-3.3)**	25.6 **(-6.0)**	3.7 **(-3.3)**	15.6 **(-2.9)**	62.3 **(-2.8)**	31.4 (-3.0)	4.8 (+0.0)	16.9 (-0.5)	66.2 (-0.7)	38.5 (+4.2)	7.0 **(+3.8)**	21.8 **(+5.8)**	60.6 **(-2.5)**	31.3 (-1.0)	4.9 (-0.2)	17.7 (+0.3)

Table 3: **Impact of instruction modality (Text vs. TTS Voice).** Header percentages are episode shares. We report performance metrics (%) and show the gap below each score (Δ = TTS − Text). To objectively highlight the cognitive load distribution, for each model and metric, the most severe performance drop among the three dependency levels is marked in **bold red**. Conversely, any performance maintenance or gain ($\Delta \geq 0$) is marked in **bold teal**. For both tested models, TTS lowers all four metrics on AV-Critical and AV-Supportive tasks; AV-Present tasks show smaller EM decreases and gains in SR and GP.

Model	Instruction	AV-Critical (29.8%)				AV-Supportive (32.4%)				AV-Present (37.8%)				Overall (100%)			
		TM	EM	SR	GP	TM	EM	SR	GP	TM	EM	SR	GP	TM	EM	SR	GP
Gemini 3 Pro	Text (Baseline)	76.9	62.7	33.2	42.2	79.6	67.4	34.4	45.0	84.7	70.1	33.0	44.4	80.0	63.6	33.4	43.6
	TTS Voice	69.0 **(-7.9)**	52.1 **(-10.6)**	29.1 (-4.1)	39.4 (-2.8)	74.5 (-5.1)	59.9 (-7.5)	27.7 **(-6.7)**	40.3 **(-4.7)**	81.6 (-3.1)	67.3 (-2.8)	35.8 **(+2.8)**	46.4 **(+2.0)**	74.3 (-5.7)	59.1 (-4.5)	30.3 (-3.1)	41.6 (-2.0)
Qwen3-Omni	Text (Baseline)	58.0	31.6	7.0	18.5	65.1	34.4	4.8	17.4	66.9	34.3	3.2	16.0	63.1	32.3	5.1	17.4
	TTS Voice	55.7 (-2.3)	26.6 (-5.0)	6.0 (-1.0)	16.8 (-1.7)	58.8 **(-6.3)**	27.2 **(-7.2)**	1.8 **(-3.0)**	14.7 **(-2.7)**	62.6 (-4.3)	33.9 (-0.4)	3.7 **(+0.5)**	17.7 **(+1.7)**	58.7 (-4.4)	28.5 (-3.8)	3.8 (-1.3)	16.2 (-1.2)

4.4 Error Analysis

Figure 3 illustrates two recurring failures of Gemini 3.0 Pro. In T0595, the model neglects the required audio pause, causing both Type Match and Exact Match failures. In T0416, it predicts the correct action type but incorrect subtitle coordinates, causing an Exact Match failure. Appendix E provides the complete case analyses.

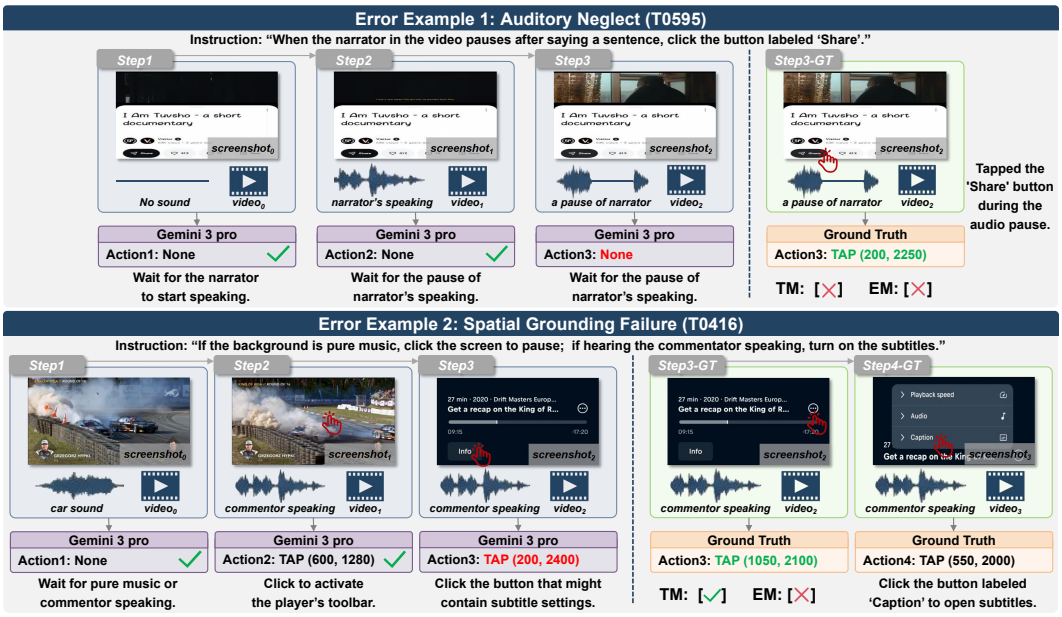


Figure 3: **Qualitative error analysis of Gemini 3.0 Pro.** (Top) *Auditory Neglect*: The model fails to trigger an action in response to a transient acoustic state change (a pause in narration). (Bottom) *Spatial Grounding Failure*: The model correctly identifies the target action primitive (`TAP`) based on the multimodal context but fails to predict the precise spatial coordinates of the subtitle icon. Coordinates shown in the figure are in original screenshot pixels rather than the normalized evaluation space.

5 Conclusion and Future Work

We introduced OmniGUI, the first step-level benchmark for GUI agents receiving continuous, interleaved images, temporal video clips, and synchronous audio at every action step in smartphone environments. It extends screenshot-based evaluation with 709 expert-demonstrated episodes (2,579 steps) spanning five cognitive dimensions and three objective multimodal dependency levels.

Our initial baselines use foundational omni-modal models as agent proxies. The results show higher EM on static Localization than on Temporal Reasoning and Cross-modal Discrimination tasks. Ablations identify cross-modal interference and performance degradation when processing dual-audio streams.

While OmniGUI establishes a comprehensive evaluation foundation, our current protocol employs an offline, step-level methodology using expert-demonstrated action histories. While this optimally isolates per-step perception-to-action capabilities and ensures deterministic reproducibility, it does not evaluate an agent's ability to recover from compounding errors during autonomous, end-to-end rollouts. Future work could explore extending the evaluation protocol to include autonomous, interactive settings to assess these dynamic error-recovery mechanisms.

AI USE STATEMENT

We evaluate generative foundation models as the systems under study, using the prompts and settings described in Section 4.1 and Appendix A. We also use text-to-speech synthesis to create spoken versions of task instructions for the instruction-modality ablation (Section 4.3). No additional generative-AI assistance was used in the original research. The authors take responsibility for the final content of this work, including its claims and artifacts.

ETHICS STATEMENT

OmniGUI is an offline benchmark built from expert demonstrations of smartphone interactions, as described in Section 3.3. The authors have identified no ethical or licensing issues associated with the reported research. The benchmark is intended to support the evaluation of multimodal GUI agents; its offline results do not establish the safety or reliability of autonomous deployment.

REPRODUCIBILITY STATEMENT

Section 3 describes the observation and action spaces, task taxonomy, collection and annotation procedures, and evaluation metrics. Appendix A provides the unified prompts, modality-ablation configurations, and reported generation settings. Appendix B documents application-level dataset statistics and illustrative data formats. Tables 1, 2, and 3 report the complete numerical results. The complete dataset, evaluation pipeline, and baseline prompts are provided in the supplementary material for peer review. The full dataset will be publicly released upon publication.

REFERENCES

Dongping Chen, Yue Huang, Siyuan Wu, Jingyu Tang, Liuyi Chen, Yilin Bai, Zhigang He, Chenlong Wang, Huichi Zhou, Yiqiang Li, et al. Gui-world: A video benchmark and dataset for multimodal gui-oriented understanding. *arXiv preprint arXiv:2406.10819*, 2024.

Kanzhi Cheng, Qiushi Sun, Yougang Chu, Fangzhi Xu, Li YanTao, Jianbing Zhang, and Zhiyong Wu. Seeclick: Harnessing gui grounding for advanced visual gui agents. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 9313–9332, 2024.

Sanjoy Chowdhury, Sayan Nag, Subhrajyoti Dasgupta, Yaoting Wang, Mohamed Elhoseiny, Ruohan Gao, and Dinesh Manocha. AVTrustBench: Assessing and Enhancing Reliability and Robustness in Audio-Visual LLMs, 2025. URL https://arxiv.org/abs/2501.02135.

Gheorghe Comanici, Eric Bieber, Mike Schaekermann, Ice Pasupat, Noveen Sachdeva, Inderjit Dhillon, Marcel Blistein, Ori Ram, Dan Zhang, Evan Rosen, et al. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. *arXiv preprint arXiv:2507.06261*, 2025.

Xiang Deng, Yu Gu, Boyuan Zheng, Shijie Chen, Sam Stevens, Boshi Wang, Huan Sun, and Yu Su. Mind2web: Towards a generalist agent for the web. *Advances in Neural Information Processing Systems*, 36:28091–28114, 2023.

Chaoyou Fu, Haojia Lin, Zuwei Long, Yunhang Shen, Yuhang Dai, Meng Zhao, Yi-Fan Zhang, Shaoqi Dong, Yangze Li, Xiong Wang, et al. Vita: Towards open-source interactive omni multimodal llm. *arXiv preprint arXiv:2408.05211*, 2024.

Chaoyou Fu, Yuhan Dai, Yongdong Luo, Lei Li, Shuhuai Ren, Renrui Zhang, Zihan Wang, Chenyu Zhou, Yunhang Shen, Mengdan 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*, pp. 24108–24118, 2025a.

Chaoyou Fu, Haojia Lin, Xiong Wang, Yi-Fan Zhang, Yunhang Shen, Xiaoyu Liu, Yangze Li, Zuwei Long, Heting Gao, Ke Li, et al. VITA-1.5: Towards GPT-4o Level Real-Time Vision and Speech Interaction. *arXiv preprint arXiv:2501.01957*, 2025b.

Google Gemini Team. Gemini 3: A Family of Highly Capable Multimodal Models. Technical report, Google DeepMind, 2025. URL https://deepmind.google/technologies/gemini/. Technical Report.

Wenyi Hong, Weihan Wang, Qingsong Lv, Jiazheng Xu, Wenmeng Yu, Junhui Ji, Yan Wang, Zihan Wang, Yuxiao Dong, Ming Ding, et al. Cogagent: A visual language model for gui agents. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 14281–14290, 2024.

Aaron Hurst, Adam Lerer, Adam P Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, et al. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*, 2024.

Lawrence 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. *arXiv preprint arXiv:2410.19100*, 2024.

Yadong Li, Jun Liu, Tao Zhang, Song Chen, Tianpeng Li, Zehuan Li, Lijun Liu, Lingfeng Ming, Guosheng Dong, Da Pan, et al. Baichuan-omni-1.5 technical report. *arXiv preprint arXiv:2501.15368*, 2025.

Yizhi Li, Yinghao Ma, Ge Zhang, Ruibin Yuan, Kang Zhu, Hangyu Guo, Yiming Liang, Jiaheng Liu, Zekun Wang, Jian Yang, et al. Omnibench: Towards the future of universal omni-language models. *arXiv preprint arXiv:2409.15272*, 2024.

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 *NeurIPS*, 2024.

Quanfeng Lu, Wenqi Shao, Zitao Liu, Lingxiao Du, Fanqing Meng, Boxuan Li, Botong Chen, Siyuan Huang, Kaipeng Zhang, and Ping Luo. Guiodyssey: A comprehensive dataset for cross-app gui navigation on mobile devices. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 22404–22414, 2025.

Christopher Rawles, Alice Li, Daniel Rodriguez, Oriana Riva, and Timothy Lillicrap. Androidinthewild: A large-scale dataset for android device control. *Advances in Neural Information Processing Systems*, 36:59708–59728, 2023.

Christopher Rawles, Sarah Clinckemaillie, Yifan Chang, Jonathan Waltz, Gabrielle Lau, Marybeth Fair, Alice Li, William Bishop, Wei Li, Folawiyo Campbell-Ajala, et al. Androidworld: A dynamic benchmarking environment for autonomous agents. *arXiv preprint arXiv:2405.14573*, 2024.

Sakshi Sakshi, Utkarsh Tyagi, Sonal Kumar, Ashish Seth, Ramaneswaran Selvakumar, Oriol Nieto, Ramani Duraiswami, Sreyan Ghosh, and Dinesh Manocha. Mmau: A massive multi-task audio understanding and reasoning benchmark. *arXiv preprint arXiv:2410.19168*, 2024.

Enxin Song, Wenhao Chai, Weili Xu, Jianwen Xie, Yuxuan Liu, and Gaoang Wang. Video-mmlu: A massive multi-discipline lecture understanding benchmark. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 6099–6113, 2025.

Gemini Team, Petko Georgiev, Ving Ian Lei, Ryan Burnell, Libin Bai, Anmol Gulati, Garrett Tanzer, Damien Vincent, Zhufeng Pan, Shibo Wang, et al. Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context. *arXiv preprint arXiv:2403.05530*, 2024.

Shuting Wang, Jiejun Tan, Zhicheng Dou, and Ji-Rong Wen. OmniEval: An Omnidirectional and Automatic RAG Evaluation Benchmark in Financial Domain, 2025. URL https://arxiv.org/abs/2412.13018.

Tianbao Xie, Danyang Zhang, Jixuan Chen, Xiaochuan Li, Siheng Zhao, Ruisheng Cao, Toh J Hua, Zhoujun Cheng, Dongchan Shin, Fangyu Lei, et al. Osworld: Benchmarking multimodal agents for open-ended tasks in real computer environments. *Advances in Neural Information Processing Systems*, 37:52040–52094, 2024.

Jin Xu, Zhifang Guo, Hangrui Hu, Yunfei Chu, Xiong Wang, Jinzheng He, Yuxuan Wang, Xian Shi, Ting He, Xinfa Zhu, Yuanjun Lv, Yongqi Wang, Dake Guo, He Wang, Linhan Ma, Pei Zhang, Xinyu Zhang, Hongkun Hao, Zishan Guo, Baosong Yang, Bin Zhang, Ziyang Ma, Xipin Wei, Shuai Bai, Keqin Chen, Xuejing Liu, Peng Wang, Mingkun Yang, Dayiheng Liu, Xingzhang Ren, Bo Zheng, Rui Men, Fan Zhou, Bowen Yu, Jianxin Yang, Le Yu, Jingren Zhou, and Junyang Lin. Qwen3-Omni Technical Report, 2025. URL https://arxiv.org/abs/2509.17765.

Yudong Yang, Jimin Zhuang, Guangzhi Sun, Changli Tang, Yixuan Li, Peihan Li, Yifan Jiang, Wei Li, Zejun Ma, and Chao Zhang. Audio-centric video understanding benchmark without text shortcut. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 6580–6598, 2025.

Yuan Yao, Tianyu Yu, Ao Zhang, Chongyi Wang, Junbo Cui, Hongji Zhu, Tianchi Cai, Haoyu Li, Weilin Zhao, Zhihui He, et al. MiniCPM-V: A GPT-4V Level MLLM on Your Phone. *arXiv preprint arXiv:2408.01800*, 2024.

Keen You, Haotian Zhang, Eldon Schoop, Floris Weers, Amanda Swearngin, Jeffrey Nichols, Yinfei Yang, and Zhe Gan. Ferret-ui: Grounded mobile ui understanding with multimodal llms. In *European Conference on Computer Vision*, pp. 240–255. Springer, 2024.

Chi Zhang, Zhao Yang, Jiaxuan Liu, Yanda Li, Yucheng Han, Xin Chen, Zebiao Huang, Bin Fu, and Gang Yu. Appagent: Multimodal agents as smartphone users. In *Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems*, pp. 1–20, 2025.

Boyuan Zheng, Boyu Gou, Jihyung Kil, Huan Sun, and Yu Su. Gpt-4v (ision) is a generalist web agent, if grounded. *arXiv preprint arXiv:2401.01614*, 2024.

A Experimental Details

This section details the exact prompt templates, user message structures, and hyperparameter configurations used in our evaluations. All evaluated models share an identical prompt structure without any model-specific prompt engineering.

A.1 Unified Prompt Templates and Input Structure

Prompt Design and Input Structure. To evaluate perception-to-action capabilities without agent-specific prompt engineering, we adopt a unified prompt consisting of a system instruction and a user message. The system prompt defines the Android GUI agent persona, specifies the complete action space (12 action primitives plus a wait/observe option), establishes the normalized $[0, 1000] \times [0, 1000]$ coordinate system, and strictly enforces a single JSON object as the output format. The user message structures the step-level context using an interleaved multimodal sequence. It sequentially presents the historical screenshot from step $t-2$ (if available), the current-step video clip, the synchronous environment audio, the current static screenshot, and the text-based task goal. To maintain ecological validity, the textual task instruction is adaptively provided in either Chinese or English, matching the native language of the target application. The ground-truth action history is provided as a structured text list of previously executed action types and parameters.

The baseline evaluation utilizes a standardized two-part prompt structure: a System Prompt and an interleaved User Message.

System Prompt

```
You are an Android GUI Agent. You are given a task and your action history,
with screenshots, audio and video. You need to perform the next action to
complete the task.

## Output Format
Output ONLY a single JSON object. No markdown, no code blocks, no explanation.
Your ENTIRE response must be valid JSON and nothing else.
You MUST ALWAYS output exactly one JSON object. NEVER output an empty response.

## Action Space
| action_type | Name   | Parameters |
|-------------|------------------|-------------------------------------------
-----|
| -1   | NONE   | (none) - Wait/observe without action   |
| 0   | TAP   | coordinate: [x, y]   |
| 1   | DOUBLE_TAP   | coordinate: [x, y]   |
| 2   | LONG_PRESS   | coordinate:[x, y]   |
| 3   | SWIPE_UP   | start_coordinate: [x, y], end_coordinate: [x, y] |
| 4   | SWIPE_DOWN   | start_coordinate: [x, y], end_coordinate:[x, y] |
| 5   | SWIPE_LEFT   | start_coordinate: [x, y], end_coordinate: [x, y] |
| 6   | SWIPE_RIGHT  | start_coordinate: [x, y], end_coordinate: [x, y] |
| 7   | INPUT   | text: "string"   |
| 8   | BACK   | (none)   |
| 9   | HOME   | (none)   |
| 10   | TASK_COMPLETE  | (none)   |
| 11   | TASK_IMPOSSIBLE | (none)   |

## Coordinate System
- Format:[x, y] where x=horizontal (left->right), y=vertical (top->bottom)
- Range: 0-1000 (normalized).[0, 0] = top-left,[1000, 1000] = bottom-right.

## Rules
- Base actions on the current screenshot (the last image in your input).
- If the task appears COMPLETE or already done, you MUST output {"action_type":
10}.
- If the task requires waiting/observing, output {"action_type": -1}.
```

```
- If the task is impossible, output {"action_type": 11}.
- Even when unsure, you MUST still output a valid JSON action. An empty response
  is NEVER acceptable.

## Examples
- Wait/observe: {"action_type": -1}
- Tap: {"action_type": 0, "coordinate": [500, 500]}
- Swipe down: {"action_type": 4, "start_coordinate":[500, 300],
  "end_coordinate": [500, 700]}
- Input text: {"action_type": 7, "text": "hello"}
- Back: {"action_type": 8}
- Task done: {"action_type": 10}
```

The **User Message** strictly follows an interleaved multimodal sequence, combining historical visual context with current-step multimodal signals. To maintain ecological validity, the `{task_description}` dynamically injects either the English or Chinese instruction corresponding to the target application's native environment.

User Message Structure

```
[1] (optional) Text:  "[History Screenshot]:"
[2] (optional) Image: <history screenshot from step t-2>
[3] (optional) Video: <current step video clip>
[4] (optional) Audio: <current step environment audio>
[5]   Image: <current screenshot>
[6]   Text:  <prompt text --- see below>

--- Prompt Text ---
Goal: {task_description}

Step: {current_step_index}

Action History:
Step 0: {action_text_0}
Step 1: {action_text_1}
...
Step N: {action_text_N}

Analyze the current screenshot and output the next action as JSON.
```

Ablation Configurations. For the ablation experiments, the input structures are deterministically modified to control specific variables. These modifications are strictly applied at the input level.

For the modality ablation experiments (Section 4.3), the respective media payloads (audio, video, or both) are physically omitted from the User Message. Concurrently, the first sentence of the System Prompt is minimally adjusted to reflect the available modalities:

System Prompt Modifications (Modality Ablation)

```
[w/o Audio]
You are an Android GUI Agent. You are given a task and your action history,
with screenshots and video. You need to perform the next action...

[w/o Video]
You are an Android GUI Agent. You are given a task and your action history,
with screenshots and audio. You need to perform the next action...
```

```
[w/o Audio & Video]
You are an Android GUI Agent. You are given a task and your action history,
with screenshots. You need to perform the next action...
```

To evaluate cognitive load during dual-audio processing (Text vs. TTS Voice), a `.wav` audio file containing the spoken instruction is injected into the User Message immediately preceding the text prompt. The textual `{task_description}` is replaced by a static placeholder, and the System Prompt is modified to include a listening directive:

Prompt Modifications (TTS Voice Instruction)

```
[System Prompt Adjustment]
Listen to the voice instruction audio for the task description.
You are an Android GUI Agent. You are given a task and your action history,
with screenshots, audio and video. You need to perform the next action...

[User Message Adjustment]
...
[4] (optional) Audio: <current step environment audio>
[4.1]          Audio: <TTS spoken instruction .wav file>
[5]   Image: <current screenshot>
[6]   Text:  <prompt text --- see below>

--- Prompt Text ---
Goal: [Please listen to the voice instruction]

Step: {current_step_index}
...
```

A.2 Model Configurations and Hyperparameters

Implementation Details. Model-specific adaptations are strictly limited to API-level payload formatting. To minimize sampling variance and obtain the models' most confident decision boundaries, we employ deterministic greedy decoding by setting the generation temperature to 0.0 and `do_sample = False` across all frameworks where explicit parameter control is supported. Where explicitly configured, the maximum generation limit is 4096 tokens.

To ensure deterministic, reproducible outputs and to evaluate the models' most confident decision boundaries, we enforced greedy decoding strategies across all evaluations. Table 4 details the generation hyperparameters used for each model. For proprietary API models, parameters were explicitly set to zero where supported. For certain open-source models deployed via default server interfaces (e.g., VITA, Baichuan-Omni-1.5), evaluations were conducted strictly under their officially recommended deterministic inference configurations.

B Dataset Construction Details

Task Formulation and Annotator Demographics. To operationalize the top-down taxonomy established in Section 3.2, we recruited 10 native smartphone users, each with over five years of daily Android operating experience. Guided strictly by the five predefined cognitive dimensions, these experienced annotators ideated and formulated goal-oriented usage scenarios across 29 diverse applications. This protocol ensures that the dataset achieves systematic theoretical coverage while maintaining authentic ecological validity.

Demonstration Recording. For each formulated task, the expert annotators executed the intended trajectory on physical Android devices. A background logging system synchronously captured the screen video at 30 frames per second (FPS), the internal device audio, and the precise touch interaction events. These 709 recorded human demonstrations serve as the optimal ground-truth trajectories

Table 4: Hyperparameter configurations for all evaluated models.

Model	Temperature	Max Tokens	Top_p	Top_k	Do_Sample	Seed
Gemini 3.0 Pro	0.0	4096	-	-	-	-
Gemini 3.0 Flash	0.0	4096	-	-	-	-
Gemini 2.5 Pro	0.0	4096	-	-	-	-
Gemini 2.5 Flash	0.0	4096	-	-	-	-
Qwen3-Omni	0.0	4096	-	-	-	-
MiniCPM-o 4.5	0.0	4096	-	-	`False`	-
VITA-1.5	Server Default	Server Default	-	-	-	-
Baichuan-Omni-1.5	Server Default	Server Default	-	-	-	-

for our evaluation. Screenshots I_t were extracted at the exact timestamp preceding each human action a_t. The video clip V_t and audio segment A_t for each step were segmented using the interval between the completion of a_{t-1} and the initiation of a_t.

Formalized Annotation. We developed a dedicated web-based annotation platform for multimodal GUI tasks. Annotators utilized this platform to transcribe the raw touch events into the formalized action space $\mathcal{A}$. For positional and gestural actions, annotators verified the target UI elements and bounded the normalized coordinates. For text inputs, the exact alphanumeric strings were recorded. Finally, each episode was assigned its objective multimodal dependency label as defined in Section 3.2.

This section provides detailed dataset statistics for the OmniGUI benchmark. Table 5 presents a comprehensive application-level breakdown, detailing the exact distribution of episodes and action steps across the five task dimensions and the three multimodal dependency levels for each of the 29 evaluated smartphone applications.

Table 5: **Comprehensive breakdown of the OmniGUI dataset per application.** The table reports the volume of episodes (Ep.) and total steps (Stp.) alongside their exact distribution across the five Task Dimensions (*Loc.* = Localization, *Sem.* = Semantic Understanding, *CrM.* = Cross-modal Discrimination, *Tmp.* = Temporal Reasoning, *Ins.* = Instant Response) and the three Multimodal Dependency levels (*Cri.* = AV-Critical, *Sup.* = AV-Supportive, *Pre.* = AV-Present).

Chinese Applications (ZH)										
App name	**Ep.**	**Stp.**	**Task dimensions**					**Modality dep.**		
			Loc	**Sem**	**CrM**	**Tmp**	**Ins**	**Cri**	**Sup**	**Pre**
Bilibili	29	91	7	5	6	6	5	10	15	4
Douyin	29	97	7	5	6	6	5	12	8	9
Meituan	26	103	6	5	6	5	4	4	5	17
QQ Music	25	95	5	5	5	5	5	1	11	13
PDD	25	72	5	5	5	5	5	5	9	11
DiDi	25	92	5	5	5	5	5	2	5	18
JD	25	80	5	5	5	5	5	3	9	13
Weibo	25	106	5	5	5	5	5	7	12	6
WeChat	25	84	7	5	4	5	4	9	8	8
Amap	25	64	5	5	5	5	5	1	3	21
RedNote	25	99	5	5	5	5	5	11	6	8
Kuaishou	24	122	4	5	5	6	4	11	6	7
Taobao	21	78	5	5	2	4	5	2	4	15
iQIYI	20	56	0	5	0	12	3	2	7	11
Alipay	14	64	5	0	5	4	0	4	1	9

English Applications (EN)										
App name	**Ep.**	**Stp.**	**Task dimensions**					**Modality dep.**		
			Loc	**Sem**	**CrM**	**Tmp**	**Ins**	**Cri**	**Sup**	**Pre**
Duolingo	35	133	5	5	5	15	5	16	11	8
Vimeo	25	86	5	5	5	5	5	12	6	7
TED	25	115	5	5	5	5	5	16	6	3
Snapchat	25	104	5	5	5	5	5	8	11	6
Spotify	25	91	5	5	5	5	5	7	12	6
Tasty	25	69	5	5	10	5	0	18	5	2
GTrans	25	88	5	5	5	5	5	11	14	0
X	25	61	5	5	5	5	5	5	8	12
TikTok	25	77	5	5	5	5	5	1	9	15
YouTube	25	123	5	5	5	5	5	3	12	10
RedBull	24	110	5	4	5	5	5	14	5	5
Amazon	22	69	5	5	4	4	4	1	4	17
IMDb	20	95	5	4	4	3	4	10	9	1
Insta	20	55	4	4	4	1	7	5	9	6

TOTAL (ALL 29 APPS): 709 Episodes, 2579 Steps.
Task Dims: Loc(145), Sem(137), CrM(141), Tmp(156), Ins(130).
Dep: Cri(211), Sup(230), Pre(268).

B.1 Data Format Example

The OmniGUI dataset is hierarchically organized by application to facilitate structured access and reproducible evaluations. For each application, the dataset separates episode-level metadata from step-level multimodal assets and execution traces.

1. Directory Structure. Listing 1 illustrates the standard directory hierarchy for a given application (e.g., `TED`). The episode-level metadata is distributed across five `.jsonl` files, corresponding to the cognitive task dimensions. The `media` directory encapsulates individual episode folders, which store the atomic step-level data including interleaved video clips, audio tracks, screenshots, and the step-wise action trace JSON files. The `dataset_TED.json` file serves as a global index, compiling all step-level annotations into a single array to streamline batch dataloading; its internal schema is identical to the step-level traces detailed in Listing 3.

Listing 1: Standardized application directory hierarchy

```
[App_Name]/ (e.g., TED/)
|-- localization.jsonl
|-- semantic_understanding.jsonl
|-- cross_modal_discrimination.jsonl
|-- temporal_reasoning.jsonl
|-- instant_response.jsonl
|-- dataset_TED.json  // Global index aggregating all steps
\-- media/
    |-- T009/
    |   |-- 1.mp4           // Step 1 video clip
    |   |-- 1.wav           // Step 1 audio track
    |   |-- 1.png           // Step 1 observation screenshot
    |   |-- 2.mp4
    |   |-- ...
    |   \-- T009.json       // Step-level action trace array
    |-- T019/
    \-- ...
```

2. Episode-Level Metadata (.jsonl). The `.jsonl` files store the task descriptions for each episode. Listing 2 presents a snippet from the `TED` application. These bilingual instructions map directly to the `{task_description}` placeholder in the unified evaluation prompt.

Listing 2: Episode metadata format (Snippet from temporal_reasoning.jsonl)

```
{"ID": "T0547", "app": "TED", "instruction_zh": "视频开始播放标志性的红白‘水
滴/宇宙’展开动画，
   当动画瞬间结束切入真实舞台场景的时刻，立即点击暂停视频。",
 "instruction_en": "As the video starts with the iconic red and white
 drop/universe animation, tap pause the moment the animation ends."}
{"ID": "T0548", "app": "TED", "instruction_zh": "观察画面与声音，当第二位主讲人
开口发声的
   一瞬间，立刻点击暂停视频。",
 "instruction_en": "Observe the video and audio, and tap pause the instant
 the second speaker begins to speak."}
{"ID": "T0551", "app": "TED", "instruction_zh": "点击首页下方正在播放的视频，当
视频瞬间放大进入全屏播放的时刻，
   立即下滑退出视频。",
 "instruction_en": "Tap the currently playing video at the bottom of the Home
screen, and
 the moment it expands to full screen, swipe down immediately to exit."}
```

3. Step-Level Execution Trace (.json). Inside each episode's media folder, a specific JSON file logs the chronological action sequence. Listing 3 presents an annotated, abbreviated snippet from an AV-Critical episode, demonstrating how waiting states (`NONE`) and execution states (`TAP`) are recorded alongside precise target bounding boxes.

Listing 3: Step-level execution trace (Snippet from T0540.json)

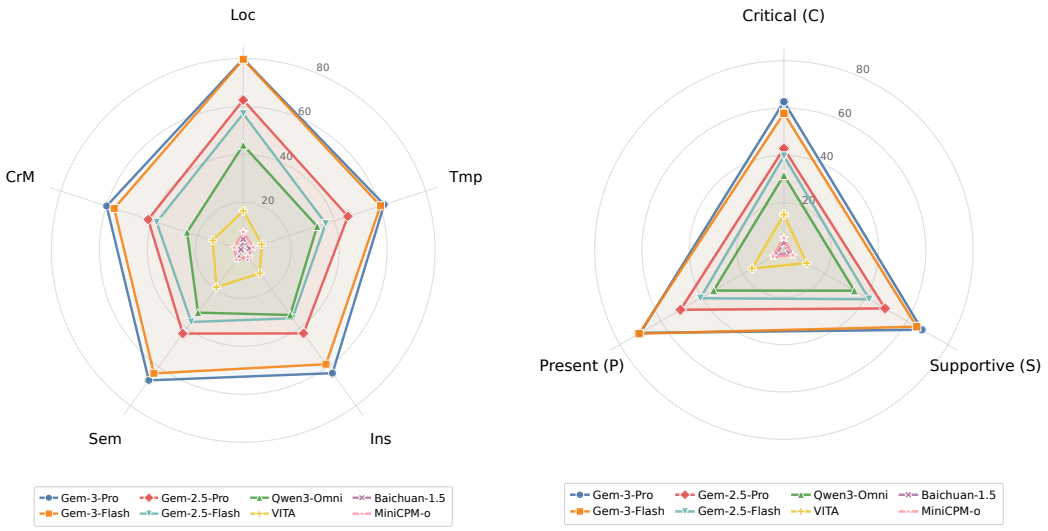
```
// 1. Episode Metadata (from JSONL)
{
 "ID": "T0540",
 "app": "TED",
 "instruction_zh": "闭屏听声音。如果听到‘观众大笑’则点击‘Share（分享）’按钮，
   如听到‘鼓掌’，则点击‘like（喜欢）’按钮,",
 "instruction_en": "Listen to the audio with the screen off. If you hear
 "Laughter", tap the "Share" button; if you hear "Applause", tap "Like"."
}
// 2. Step-level Array (Truncated for display)
[
  {
    "episode_id": "T0540",
    "episode_length": 7,
    "step_id": 0,
    "instruction_en": "Listen to the audio with the screen off...",
    "video_path": "T0540/1.mp4",
    "audio_path": "T0540/1.wav",
    "image_path": "T0540/1.png",
    "image_width": 720,
    "image_height": 1600,
    "result_action_type": -1,       // Mapped to Action Space: NONE (Wait)
    "result_action_text": "",
    "result_touch_xy": "",
    "result_lift_xy": "",
    "result_action_json": "{\"action_type\":-1,\"text\":\"\",\"touch_xy\":\"\"}"
  },
  ... // Steps 1 to 4 omitted (waiting states with continuous audio input)
  {
    "episode_id": "T0540",
    "episode_length": 7,
    "step_id": 5,
    "instruction_en": "Listen to the audio with the screen off...",
    "video_path": "T0540/6.mp4",
    "audio_path": "T0540/6.wav",
    "image_path": "T0540/6.png",
    "image_width": 720,
    "image_height": 1600,
    "result_action_type": 0,        // Mapped to Action Space: TAP
    "result_action_text": "",
    "result_touch_xy": "[[210,786],[351,850]]", // Target bounding box
    "result_lift_xy": "",
    // Long string manually wrapped for display formatting
    "result_action_json": "{\"action_type\":0,\"text\":\"\",
                           \"touch_xy\":\"[[210,786],[351,850]]\"}"
  }
]
```

C Additional Experimental Results

This section provides supplementary visual analyses of the baseline evaluations, offering a more granular perspective on model capabilities across different cognitive dimensions, multimodal dependencies, and application environments.

C.1 Model Capability Fingerprints

To visualize the specific performance profiles of the evaluated models, Figure 4 presents multidimensional capability fingerprints using the Exact Match (EM) metric. The analysis is divided into two perspectives: performance across the five predefined task dimensions and performance across the three multimodal dependency levels.



(a) Five task dimensions

(b) Three dependency levels

Figure 4: **Capability fingerprints of evaluated models.** The radar charts map the Exact Match (EM) performance. (a) Across the five operational dimensions, models universally exhibit stronger capabilities on static Localization (Loc) compared to Temporal Reasoning (Tmp) and Cross-modal Discrimination (CrM). (b) The dependency-level profiles show how EM varies across AV-Present (P), AV-Supportive (S), and AV-Critical (C) tasks for each model.

C.2 Performance Breakdown by Application

To illustrate the variance in execution difficulty across different smartphone interfaces, Figure 5 details the performance of the strongest baseline model, Gemini 3.0 Pro, disaggregated by the 29 evaluated applications.

The horizontal bar chart reports the Exact Match (EM), Goal Progress (GP), and Success Rate (SR) metrics, sorted in ascending order by GP from bottom to top. The overall benchmark averages (EM = 63.6%, GP = 43.6%) are indicated by vertical dotted lines. The variance across applications objectively underscores the diversity of GUI complexities captured within OmniGUI.

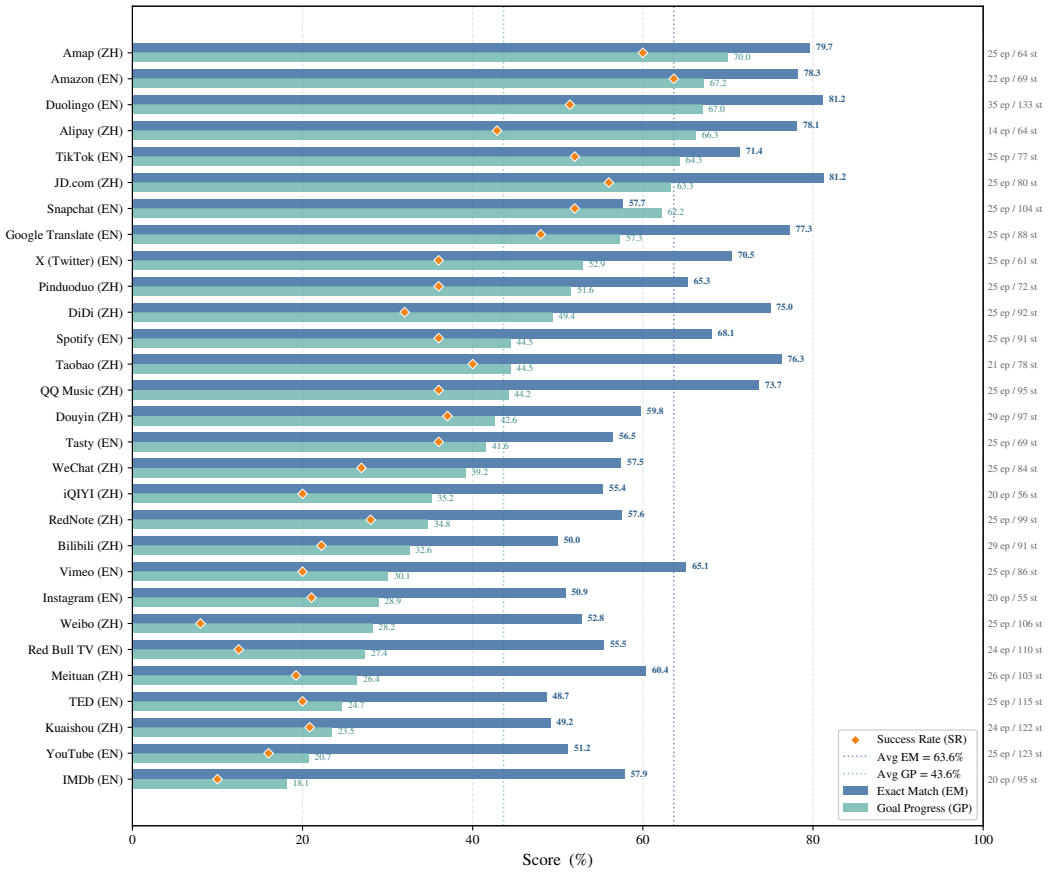


Figure 5: **Gemini 3.0 Pro performance disaggregated by application.** The applications are sorted by Goal Progress (GP) scores. The right-hand axis explicitly denotes the sample volume (episodes/steps) for each application. The vertical dotted lines represent the overall benchmark averages, providing a reference to identify application environments that present above-average or below-average difficulty. The labeled EM and GP scores are reproduced at a larger text size in Table 6.

Table 6: Labeled application-level scores from Figure 5. EM/GP (%); episode/step counts match the original figure. SR remains shown in the source figure.

Application	Episodes	Steps	EM (%)	GP (%)
Amap (ZH)	25	64	79.7	70.0
Amazon (EN)	22	69	78.3	67.2
Duolingo (EN)	35	133	81.2	67.0
Alipay (ZH)	14	64	78.1	66.3
TikTok (EN)	25	77	71.4	64.3
JD.com (ZH)	25	80	81.2	63.3
Snapchat (EN)	25	104	57.7	62.2
Google Translate (EN)	25	88	77.3	57.3
X (Twitter) (EN)	25	61	70.5	52.9
Pinduoduo (ZH)	25	72	65.3	51.6
DiDi (ZH)	25	92	75.0	49.4
Spotify (EN)	25	91	68.1	44.5
Taobao (ZH)	21	78	76.3	44.5
QQ Music (ZH)	25	95	73.7	44.2
Douyin (ZH)	29	97	59.8	42.6
Tasty (EN)	25	69	56.5	41.6
WeChat (ZH)	25	84	57.5	39.2
iQIYI (ZH)	20	56	55.4	35.2
RedNote (ZH)	25	99	57.6	34.8
Bilibili (ZH)	29	91	50.0	32.6
Vimeo (EN)	25	86	65.1	30.1
Instagram (EN)	20	55	50.9	28.9
Weibo (ZH)	25	106	52.8	28.2
Red Bull TV (EN)	24	110	55.5	27.4
Meituan (ZH)	26	103	60.4	26.4
TED (EN)	25	115	48.7	24.7
Kuaishou (ZH)	24	122	49.2	23.5
YouTube (EN)	25	123	51.2	20.7
IMDb (EN)	20	95	57.9	18.1

D Benchmark Reference Tables

Tables 7 and 8 provide the complete action-space definitions and comparison with prior GUI benchmarks referenced in the main text.

Table 7: **Definition of the OmniGUI Action Space.** The agent predicts an action primitive along with its parameters. Coordinates (x, y) are normalized to the resolution-independent range $[0, 1000]^2$.

Action Primitive	Parameter Space	Semantics / Usage
Temporal / Idle		
`NONE` (-1)	$\varnothing$	Wait or observe without taking action
Positional Actions		
`TAP` (0)	$(x, y) \in [0, 1000]^2$	Click a UI element (button, link, icon)
`DOUBLE_TAP` (1)	$(x, y) \in [0, 1000]^2$	Trigger specific states (e.g., zoom, like)
`LONG_PRESS` (2)	$(x, y) \in [0, 1000]^2$	Open context menus or select items
Gestural Actions		
`SWIPE_UP` (3)	$(x, y) \in [0, 1000]^2$	Scroll down content or feed
`SWIPE_DOWN` (4)	$(x, y) \in [0, 1000]^2$	Refresh page or scroll up
`SWIPE_LEFT` (5)	$(x, y) \in [0, 1000]^2$	Navigate carousels or switch tabs
`SWIPE_RIGHT` (6)	$(x, y) \in [0, 1000]^2$	Navigate back or switch tabs
Text Input		
`INPUT` (7)	$\mathcal{S}$ (String)	Enter text into a focused field
System & Status Signals		
`BACK` (8)	$\varnothing$	Return to previous screen/activity
`HOME` (9)	$\varnothing$	Return to device home screen
`TASK_COMPLETE` (10)	$\varnothing$	Signal successful task completion
`TASK_IMPOSSIBLE` (11)	$\varnothing$	Signal task is infeasible/stuck

Table 8: **Comparison of OmniGUI with representative GUI agent benchmarks.** *Audio* and *Video* indicate whether the benchmark provides auditory and video inputs beyond static screenshots. *Per-Step* denotes whether multimodal inputs are provided at *every* action step, rather than as pre-task or reference content. *Action Output* specifies the format of predicted actions. *Manual* indicates whether all tasks are manually designed.

Benchmark	Platform	#Tasks	#Steps	Input Modalities			Per-Step	Action Output	Manual
				Image	Video	Audio			
Vision-Only Benchmarks									
AITW (Rawles et al., 2023)	Android	30,378	715K+	✓	✗	✗	✗	Coordinate	✗
GUI-Odyssey (Lu et al., 2025)	Android	7,735	74K+	✓	✗	✗	✗	Coordinate	✗
AndroidWorld (Rawles et al., 2024)	Android	116	–	✓	✗	✗	✗	Coordinate	✓
Mind2Web (Deng et al., 2023)	Web	2,350	12K+	✓	✗	✗	✗	DOM Element	✗
OSWorld (Xie et al., 2024)	Desktop	369	–	✓	✗	✗	✗	Coordinate	✓
ScreenSpot (Cheng et al., 2024)	Multi	1,272	–	✓	✗	✗	✗	Coordinate	✓
Benchmarks with Partial Multimodal Support									
MM-Mind2Web (Zheng et al., 2024)	Web	2,000+	–	✓	✗	✓	✗	DOM Element	✗
GUI-World (Chen et al., 2024)	Multi	12,379	–	✓	✓	✗	✗	QA / Caption	✗
VideoGUI (Lin et al., 2024)	Multi	178	4K+	✓	✓	✗	✗	Coordinate	✓
VideoWebArena (Jang et al., 2024)	Web	2,021	24K–38K	✓	✓	✓	✗	DOM Element	✗
Per-Step Multimodal Benchmark									
OmniGUI (Ours)	**Android**	**709**	**2,579**	✓	✓	✓	✓	**Coordinate**	✓

E Qualitative Error Analysis Details

The following analyses expand the cases shown in Figure 3 and discussed in Section 4.4.

Auditory Neglect. Figure 3 (top) illustrates a failure on task T0595 (within the Vimeo app), where the execution timing depends strictly on a transient audio event. The instruction requires the agent to tap the "Share" button specifically when the video narrator pauses. At Step 1 and Step 2, the model correctly outputs `NONE` while the audio contains silence and continuous speech, respectively. However, at Step 3, when the required audio pause occurs, the model continues to predict `NONE` instead of the ground-truth `TAP` action. This case demonstrates an instance where the model fails to map a step-level acoustic state change to the corresponding action execution, resulting in both Type Match (TM) and Exact Match (EM) failures.

Spatial Grounding Failure. Figure 3 (bottom) depicts task T0416 (within the Red Bull TV app), which requires invoking the video toolbar and enabling subtitles upon hearing the commentator. At Step 2, the model correctly predicts a `TAP` to activate the toolbar. At Step 3, the model correctly predicts the action type (`TAP`) to interact with the subtitle settings, successfully fulfilling the TM metric. However, the predicted coordinates $(200, 2400)$ in original screenshot pixels deviate significantly from the ground-truth target location of the subtitle icon $(1050, 2100)$. This isolated EM failure indicates that while the model successfully comprehends the multimodal instruction and determines the correct operational primitive, precise spatial grounding on the complex visual interface remains a challenge.