

HiconAgent: History Context-aware Policy Optimization for GUI Agents

Supplementary Material

A. Information flow analysis

Following the setup described in Section 4, we decompose history interactions into three parts: from history images to history actions, from history images to predictions, and from history actions to predictions. To probe how historical information flows through the model, we conduct a layer-conditioned token drop analysis—dropping history tokens (V_{his} , A_{his} , or both) after layer k and observing the resulting performance.

Shallow depth ($k < 12$): As shown in Figure 3, dropping A_{his} in shallow layers causes a large performance degradation, while dropping V_{his} at the same depths is much less harmful. This pattern indicates that effective use of historical information *depend on the action tokens as the anchor*: even if rich visual information is retained, later layers cannot directly extract effective cues from V_{his} without the action anchors. By contrast, dropping V_{his} in this range is more tolerable, since the model has already merged part of the visual history into A_{his} ; the action anchors then pass that information forward through subsequent layers.

Deeper depth ($k \geq 12$): For $k > 12$, the *action-drop* curve rises quickly and, by mid depth, its gap to the *image-drop* curve is small. This indicates that, in these layers, predictions obtain historical information through interactions with the action anchors. Once that interaction has already occurred before the drop point, removing A_{his} or V_{his} later causes only little decline, and performance continues to improve with depth. When $k \geq 24$, all three curves converge to the no-compression accuracy, indicating that the model has already integrated most historical information and can proceed without retaining these history tokens.

B. Reward Design

In GUI navigation tasks, each action consists of a type and a value. The type is chosen from a set of discrete options (e.g., CLICK, SCROLL), while the value format varies depending on the type. Based on this characteristic of GUI tasks, we introduce the following three rewards:

Format reward (r^f): This term ensures the model’s response adheres to a predefined structure like `<think>...</think><answer>...</answer>`. It returns 1 if the format is correct, 0 otherwise, promoting structured reasoning and output generation.

Action type reward (r^t): We assign 1 if the predicted action type exactly matches the ground-truth type, and 0 otherwise. This term enforces correctness at the semantic level of action selection.

Action value reward (r^v): For actions *without* values

(e.g., PRESS_BACK), the reward is 1 if the type is correct. For actions with *textual* values (e.g., TYPE, OPEN_APP), we compute the F1 score between prediction and ground truth, awarding 1 if $F1 > 0.5$. For actions with *discrete* values (e.g., SCROLL), the value must exactly match the ground truth. For *coordinate-based* actions (e.g., CLICK), we calculate the Euclidean distance d between predicted and actual coordinates, and assign a continuous reward $r^v = 1 - d$ to allow fine-grained feedback for grounding accuracy.

The final reward used for policy optimization is the sum of the three components:

$$r = r^f + r^t + r^v \quad (7)$$

C. Training details

This section includes implementation details of our training and evaluation. We will release the source code to ensure the reproducibility of our results.

Metrics. We evaluate our model on three representative navigation-oriented datasets, AndroidControl-High[18], AITW[17] and GUI-Odyssey[29], considering only their test splits under an out-of-distribution (OOD) evaluation setting to assess generalization performance. We use three standard metrics widely adopted in prior work on GUI agents, following the evaluation protocol of Os-Atlas [45]. Specifically, we report action type prediction accuracy (**Type**), GUI grounding accuracy (**Grounding**), and step success rate (**SR**).

FLOPs is computed using deepspeed flops-profiler with batch size 1. We include all model components when calculating FLOPs. The reported number is averaged over 200 samples from the training set with history length $\tau = 2$.

Training and Evaluation. HiconAgent-3B is built upon Qwen2.5-VL-3B. We follow the same training setting in GUI-R1 [31]. In the reinforcement fine-tuning stage, we adopt the following hyperparameter settings to ensure stable optimization. The rollout batch size and global batch size is set to 64, with 8 rollouts per update step. We use a small learning rate of 1×10^{-6} to stabilize training. To balance computational efficiency and generation quality, the maximum number of input pixels is capped at 1,003,520. For the language input and output, both the maximum prompt length and the maximum response length are restricted to 2048 tokens. The rollout temperature is fixed at 1.0 to encourage diverse yet consistent exploration during training.

To keep the number of tokens consistent with prior work, we set the history window size to 2, meaning the agent can access up to two past interaction steps. Each historical

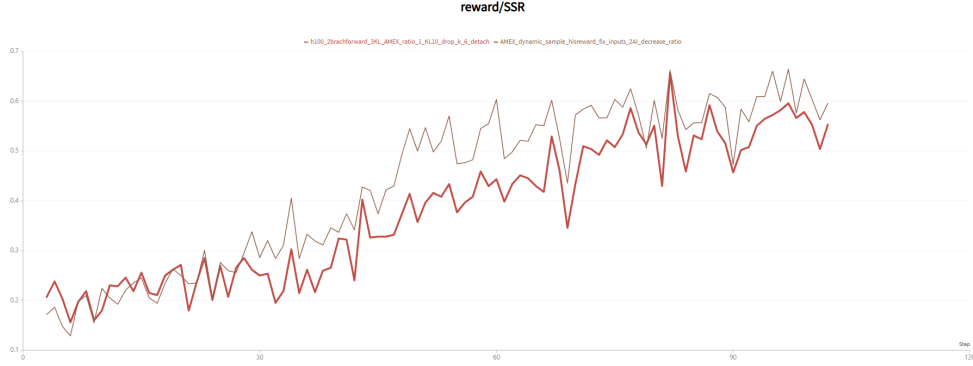


Figure 8. Training accuracy curves of Hicon-Agent with and without DCS under the AHC framework. Models trained with DCS exhibit consistently higher accuracy and faster convergence, demonstrating that adaptive history sampling facilitates more effective learning.

step includes both the screenshot observation and the corresponding action, i.e., $H_t = \{(s_{t-2}, a_{t-2}), (s_{t-1}, a_{t-1})\}$. All images are resized to a fixed resolution, and the number of visual tokens after encoding is limited to a maximum of 512 to ensure computational efficiency.

D. Visualization of SSR curve during training

During training, we monitor the execution accuracy to evaluate the learning behavior of the policy over time. As shown in Figure 8, integrating DCS into the AHC framework leads to a clear improvement in training dynamics. The curve with DCS achieves both higher convergence speed and better final accuracy compared to the AHC-only baseline. This indicates that naively relying on fixed-length histories limits the model’s ability to generalize. In contrast, DCS adaptively determines the effective history length based on task complexity, enabling the agent to access relevant information while avoiding overfitting to noisy or irrelevant historical states. These results verify that dynamically sampled history promotes more stable optimization and enhances the agent’s capability to learn meaningful sequential dependencies.

E. Visualization of sampling distribution in DCS.

We sample the number of preserved history blocks $\tau_i \in \{0, 1, 2\}$ from a time-dependent exponential distribution. At training step u , the sampling distribution is defined as

$$P(\tau_i | u) = \frac{\exp(\lambda(u) \tau_i)}{\sum_{j=0}^N \exp(\lambda(u) j)},$$

where $\lambda(u)$ is a linear function that increases with u . As shown in Figure 9, in the early stage of training, $\lambda(u) \approx 0$ and the distribution is nearly uniform, encouraging random exploration. As training progresses, $\lambda(u)$ gradually grows, yielding an increasingly biased exponential distribution that favors larger values of τ_i . This schedule smoothly shifts

the sampling strategy from random selection to full-context history.

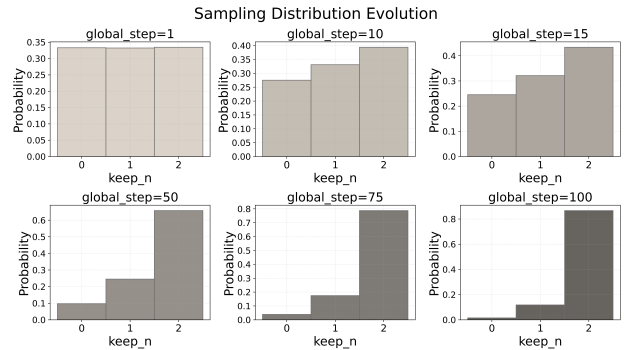


Figure 9. Evolution of the sampling distribution from uniform exploration to exponentially biased selection as training progresses.

We define $\lambda(u)$ as a function that increases from 0 to λ_{\max} within the first αT steps, gradually shifting the sampling distribution from uniform to exponentially biased:

$$\lambda(u) = \lambda_{\max} \cdot \min\left(1, \frac{u}{\alpha T}\right) \quad (8)$$

T denotes the total number of training steps. λ_{\max} controls the final steepness of the exponential bias (larger λ_{\max} yields a more strongly peaked distribution that favors larger τ_i), while α determines the fraction of training used for warm-up. In our implementation, we set $\lambda_{\max} = 2$, $\alpha = \frac{1}{3}$.

F. GUI datasets

We construct our training data from the open-source AMEX dataset[2], which contains high-level GUI interaction trajectories. Preserving the original action distribution, we randomly sample 3,000 steps without applying additional filtering or cleaning procedures.

AMEX AMEX is a large-scale mobile GUI dataset for training and evaluating control agents, comprising >104K

high-resolution screenshots from 110 Android apps with multi-level annotations. Each episode contains about 13 actions on average.

Action distribution (full dataset ratio):

- `click(start_box='(x, y)')` **24815 (64.11%)**
- `scroll(direction='down or up or right or left')` **7628 (19.71%)**
- `finished()` **2828 (7.31%)**
- `type(content='')` **2419 (6.25%)**
- `press_enter()` **651 (1.68%)**
- `impossible()` **220 (0.57%)**
- `press_back()` **135 (0.35%)**
- `press_home()` **13 (0.03%)**

AndroidControl AndroidControl is diverse benchmark to study data scaling for UI control, containing 15,283 demonstrations spanning 14,548 unique tasks across 833 Android apps, with both high-level and low-level human-written instructions for each task. Each episode contains about 5 actions on average.

Action distribution (test split):

- `click(start_box='(x, y)')` **5074 (50.81%)**
- `finished()` **1543 (15.45%)**
- `scroll(direction='down or up or right or left')` **1211 (12.13%)**
- `type(content='')` **632 (6.33%)**
- `open_app(app_name='')` **608 (6.09%)**
- `wait()` **567 (5.68%)**
- `press_back()` **343 (3.43%)**
- `long_press(start_box='(x, y)')` **9 (0.09%)**

GUI-Odyssey GUI-Odyssey is a cross-app mobile GUI navigation dataset for multi-step workflows across apps; the paper reports 7,735 episodes over 6 devices, 6 task types, 201 apps and $\sim 1.4\text{K}$ app combinations. Each episode contains about 15 actions on average.

Action distribution (test split):

- `click(start_box='(x, y)')` **19142 (65.05%)**
- `type(content='')` **3113 (10.58%)**
- `scroll(direction='down or up or right or left')` **2764 (9.39%)**
- `press_home()` **2233 (7.59%)**
- `finished()` **1875 (6.37%)**
- `long_press(start_box='(x, y)')` **106 (0.36%)**
- `press_recent()` **74 (0.25%)**
- `press_back()` **61 (0.21%)**
- `impossible()` **58 (0.20%)**

AITW Android in the Wild (AITW) is a large-scale dataset for Android device control using natural language instructions. The paper reports 715,000 episodes, 30,000+ unique

instructions, 8 device types (Pixel 2 XL through Pixel 6), 4 Android versions, covering hundreds of apps and websites. Each episode contains about 6.5 actions on average.

G. Effect of history observation and compression

We first examine the impact of historical observations by comparing Qwen2.5VL-3B(2A), which uses only past actions, with Qwen2.5VL-3B(2AO), which includes both actions and observations from the past two steps. Incorporating visual history brings clear improvements across all metrics (+8.96% SR), highlighting the importance of visual context in guiding decision-making. When applying inference-only compression to the 2AO model, performance drops significantly (-4.95% SR). In contrast, our **Hicon-Agent-3B**, trained with history-aware optimization, recovers most of this loss and improves performance(+5.06 % SR) compared to the compressed baseline. It also exceeds the uncompressed 2AO baseline in SR and grounding accuracy, demonstrating more effective and efficient usage of historical context. This highlights the advantage of our training strategy in mitigating the trade-off between computational efficiency and task performance.

Models	Hist. Comp.	Type	Ground.	SR	FLOPs	
Qwen2.5VL-3B	2A	-	61.05	59.61	43.33	13.21
Qwen2.5VL-3B	2AO	-	69.29	63.88	52.29	35.75
Qwen2.5VL-3B	2AO	✓	66.56 _(-2.73)	61.10 _(-2.78)	47.34 _(-4.95)	25.21 _(-10.54)
Hicon-Agent-3B	2AO	✓	67.79 _(+1.23)	65.01 _(+3.91)	52.40 _(+5.06)	25.21 _(-10.54)

Table 6. Study on different strategies for history utilization on the AndroidControl dataset. **Red** indicates improvement, **green** indicates drop. **Hist.** denotes the history context format, and **Comp.** indicates whether history compression is applied.

H. Algorithm details

Algorithm 1 details the HCPO training loop. We begin with on-policy, group-wise rollouts using Dynamic Context Sampling: for each of the G samples, we first construct the full input context $\{I, H_t, s_t\}$, then draw a truncated history H_t^i by sampling a history length $\tau_i \leq \tau$ sampled from the distribution p . Given this truncated context, we sample a response $o_i \sim \pi_{\theta_{\text{old}}}(\cdot | I, H_t^i, s_t)$. The corresponding reward $\{r_i\}$ is computed and converted into group advantages $\{A_i\}$.

Each sampled response is then evaluated by two forward passes of the current policy with shared parameters: (i) an uncompressed branch that uses $\{I, H_t, s_t\}$ end-to-end; and (ii) a compressed branch that mirrors the first k layers and then drops history *vision* tokens after layer k , retaining action and other tokens to form $H_t^{i,c}$. Reusing the same $\{o_i\}$ isolates the effect of compression.

We optimize two clipped-ratio gradient policy losses against π_θ , while adding a token-level consistency term that pulls the compressed distribution $\pi_\theta(\cdot | I, H_t^{i,c}, s_t)$ toward the uncompressed distribution $\pi_\theta(\cdot | I, H_t^i, s_t)$. Teacher logits from the uncompressed branch are detached to prevent gradient flow. The final objective $\mathcal{L}_{\text{HCPO}}$ preserves on-policy learning under complete history and aligns the compressed path for efficient inference.

Algorithm 1 History Context-aware Policy Optimization (HCPO)

Require: Policy model π_θ , old policy $\pi_{\theta_{\text{old}}}$, reward model R , task instruction I , current GUI observation s_t , history context $H_t = \{(s_{t-\tau}, a_{t-\tau}), \dots, (s_{t-1}, a_{t-1})\}$, group size G , compression layer k , consistency weight λ

- 1: **# Group rollout with DCS**
 - 2: Initialize sampling distribution p
 - 3: Build full history sequence: $\{I, H_t, s_t\}$
 - 4: **for** $i = 1$ to G **do**
 - 5: Sample history length $\tau_i \sim p$ to get H_t^i
 - 6: Sample and select response $o_i \sim \pi_{\theta_{\text{old}}}(\cdot | I, H_t^i, s_t)$
 - 7: **end for**
 - 8: Evaluate rewards $\{r_1, \dots, r_G\} \leftarrow R(\{o_1, \dots, o_G\})$
 - 9: Compute group-wise advantages $\{A_1, \dots, A_G\}$
 - 10: **# Full history branch forward pass**
 - 11: **for** $i = 1$ to G **do**
 - 12: Compute logits $\pi_{\theta_{\text{old}}}(o_i | I, H_t^i, s_t)$
 - 13: Compute logits $\pi_\theta(o_i | I, H_t^i, s_t)$
 - 14: Compute reference logits $\pi_{\theta_{\text{ref}}}(o_i | I, H_t^i, s_t)$
 - 15: **end for**
 - 16: **# Compressed history branch forward pass**
 - 17: **for** $i = 1$ to G **do**
 - 18: Reuse response o_i and perform forward pass with partial history compression:
 - 19: Use uncompressed input $\{I, H_t^i, s_t\}$ in first k layers
 - 20: Drop history vision tokens from H_t^i in layers $k + 1$ to get $H_t^{i,c}$
 - 21: Compute compressed logits $\pi_\theta(o_i | I, H_t^{i,c}, s_t)$
 - 22: Compute token-level KL divergence: $\mathbb{D}_{\text{KL}}(\pi_\theta(o_i | I, H_t^{i,c}, s_t) \parallel \pi_\theta(o_i | I, H_t^i, s_t))$
 - 23: **end for**
 - 24: **# Compute policy-gradient loss and consistency loss**
 - 25: Compute uncompressed policy loss $\mathcal{L}_{\text{w/o comp}}$
 - 26: Compute compressed policy loss $\mathcal{L}_{\text{w/comp}}$
 - 27: Compute consistency loss: $\mathcal{L}_{\text{KL}} = \sum_{i=1}^G \mathbb{D}_{\text{KL}}(\cdot)$
 - 28: Compute total loss: $\mathcal{L}_{\text{HCPO}} = \mathcal{L}_{\text{w/o comp}} + \mathcal{L}_{\text{w/comp}} + \lambda \mathcal{L}_{\text{KL}}$
 - 29: Update model: $\theta \leftarrow \theta - \eta \nabla_\theta \mathcal{L}_{\text{HCPO}}$
-

I. Prompts for training and evaluation

You are a GUI agent. You are given a task and your action history, with screenshots. You need to perform the next action to complete the task. You FIRST need to think based on the current image, task, and historical actions. The reasoning process MUST BE enclosed within `<think> </think>` tags. Then output the action, which MUST BE put in `<action> </action>` and MUST BE in Action Space.

```
## Output Format
<think>...</think><action>...</action>

## Action Space
click(start_box='(x,y)')
type(content='')
scroll(direction='down or up or right or left')
press_back()
press_home()
press_enter()
finished()

## Example:
<think>The user wants to search for shoes. The current screen has a search bar at the top.</think>
<action>click(start_box='(x,y)')</action>
```

Listing 1. AMEX training prompt template.

You are a GUI agent. You are given a task and your action history, with screenshots. You need to perform the next action to complete the task. You FIRST need to think based on the current image, task, and historical actions. The reasoning process MUST BE enclosed within `<think> </think>` tags. Then output the action, which MUST BE put in `<action> </action>` and MUST BE in Action Space.

```
## Output Format
<think>...</think><action>...</action>

## Action Space
click(start_box='(x,y)')
long_press(start_box='(x,y)')
type(content='')
scroll(direction='down or up or right or left')
open_app(app_name='')
press_back()
```

```

press_home()
wait()
finished()

## Example:
<think>The user wants to search for shoes. The current screen has a search bar at the top.</think>
<action>click(start_box='(x,y)')</action>

```

Listing 2. AndroidControl evaluation prompt template.

You are a GUI agent. You are given a task and your action history, with screenshots. You need to perform the next action to complete the task. You FIRST need to think based on the current image, task, and historical actions. The reasoning process MUST BE enclosed within `<think> </think>` tags. Then output the action, which MUST BE put in `<action> </action>` and MUST BE in Action Space.

```

## Output Format
<think>...</think><action>...</action>

## Action Space
click(start_box='(x,y)')
long_press(start_box='(x,y)')
type(content='')
scroll(direction='down or up or right or left')
impossible()
press_back()
press_home()
press_recent()
finished()
## Example:
<think>The user wants to search for shoes. The current screen has a search bar at the top.</think>
<action>click(start_box='(x,y)')</action>

```

Listing 3. GUI-Odyssey evaluation prompt template.

You are a GUI agent. You are given a task and your action history, with screenshots. You need to perform the next action to complete the task. You FIRST need to think based on the current image, task, and historical actions. The reasoning process MUST BE enclosed within `<think> </think>` tags. Then output the action, which MUST BE

```

put in <action> </action> and MUST BE in Action Space.

## Output Format
<think>...</think><action>...</action>

## Action Space
click(start_box='(x,y)')
long_press(start_box='(x,y)')
type(content='')
scroll(direction='down or up or right or left')
impossible()
press_enter()
press_back()
press_home()
finished()
## Example:
<think>The user wants to search for shoes. The current screen has a search bar at the top.</think>
<action>click(start_box='(x,y)')</action>

```

Listing 4. AITW evaluation prompt template.

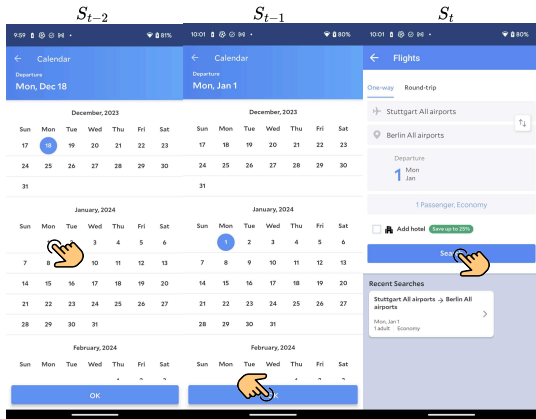
J. Model Behavior Across Different History Lengths

To better understand how history length affects agent behavior, we provide a case study comparing the base model and our HiconAgent-3B under different history lengths $\tau \in \{0, 1, 2\}$. As shown in Figure 10, the base model performs correctly when using shorter contexts ($\tau = 0$ or $\tau = 1$), but fails when the history is extended to $\tau = 2$, where the additional observations introduce distracting or misleading information, causing the model to attend to an incorrect UI element and produce the wrong action. In contrast, our model, trained with Dynamic Context Sampling, still produces the correct action when $\tau = 2$. Since DCS exposes the agent to diverse and progressively biased history lengths during optimization, the model learns to effectively utilize extended context. This qualitative evidence supports our quantitative results, demonstrating that naively increasing history is suboptimal, whereas HCPO equips the agent with robustness across variable context windows and enables it to benefit from longer history when necessary.

K. Visualization of successful trajectories

To better illustrate how our proposed HCPO framework facilitates robust decision-making, we visualize several representative successful trajectories from the evaluation benchmarks as shown in Figure 11a and Figure 11b.

Instruction: I'm going to meet my son in Berlin for important work with my wife and I decided to go there by flight, so search for the one way flight from Stuttgart to Berlin airport on January 1, 2024 in the Agoda app



Input Context: $\tau = 2 \{S_{t-2}, S_{t-1}, S_t\}$

Hicon-Agent-3B: <think>Since the goal is to search for a one-way flight from Stuttgart to Berlin, I need to click on the 'Search' button to find available flights from Stuttgart to Berlin on January 1, 2024. </think>
<action>click(start_box=(209,421))</action>

Qwen2.5VL-3B: <think>The user wants to go from Stuttgart to Berlin, the next logical step is to click on the 'One-way' option to switch the search from round-trip to one-way. </think>
<action>click(start_box=(51,151))</action>

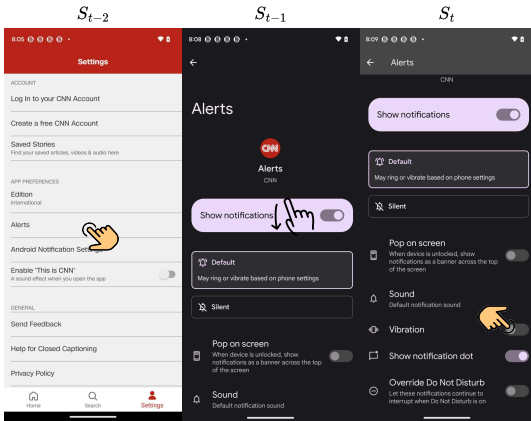
Input Context: $\tau = 1 \{S_{t-1}, S_t\}$

Qwen2.5VL-3B: <think>Since the departure and destination are already set, the next logical step is to initiate the search for flights. The 'Search' button is visible and ready to be clicked to proceed with the search. </think>
<action>click(start_box=(218,534))</action>

Input Context: $\tau = 0 \{S_t\}$

Qwen2.5VL-3B: <think>The current screen is already set up with the correct departure and destination airports. The next logical step is to initiate the search for available flights. The 'Search' button is visible and ready to be clicked. </think>
<action>click(start_box=(213, 534))</action>

Instruction: In the CNN App, turn on the breaking news alert with vibration to receive an alert in Settings.



Input Context: $\tau = 2 \{S_{t-2}, S_{t-1}, S_t\}$

Hicon-Agent-3B: <think>Since the goal is to enable the vibration alert, I need to enable the Vibration option to complete the task of setting up the alert with pop-up and vibration. </think>
<action>click(start_box=(375, 705))</action>

Qwen2.5VL-3B: <think>The user wants to turn on the breaking news alert with vibration, and the vibration option is already enabled, the task is complete. The user has successfully configured the necessary setting. </think>
<action>finished()</action>

Input Context: $\tau = 1 \{S_{t-1}, S_t\}$

Qwen2.5VL-3B: <think>The goal is to turn on the breaking news alert with vibration, I need to enable it by clicking on the toggle switch. This will allow the user to receive an alert with a vibration. </think>
<action>click(start_box=(373, 708))</action>

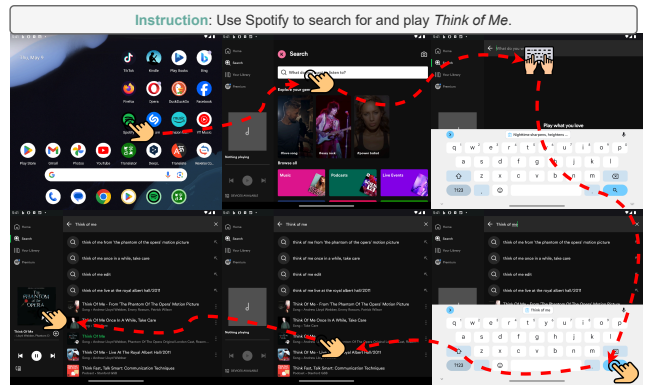
Input Context: $\tau = 0 \{S_t\}$

Qwen2.5VL-3B: <think>I need to ensure that the 'Vibration' toggle is switched on. The current screen shows the 'Vibration' toggle is off, so I need to click on it to enable it. </think>
<action>click(start_box=(375, 711))</action>

Figure 10. Case study of model behavior under different history length context.



(a) Trajectory example 1



(b) Trajectory example 2

Figure 11. Case studies of our model on downstream GUI navigation tasks.

References

[1] Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko

Alteneschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023. 2

[2] Yuxiang Chai, Siyuan Huang, Yazhe Niu, Han Xiao, Liang Liu, Dingyu Zhang, Peng Gao, Shuai Ren, and Hongsheng Li.

- AMEX: Android Multi-annotation Expo Dataset for Mobile GUI Agents. *arXiv preprint*, 2024. arXiv:2407.17490 [cs]. 6, 2
- [3] Gongwei Chen, Xinhang Song, Haitao Zeng, and Shuqiang Jiang. Scene recognition with prototype-agnostic scene layout. *IEEE Transactions on Image Processing*, 29:5877–5888, 2020. 2
- [4] Gongwei Chen, Xinhang Song, Bohan Wang, and Shuqiang Jiang. See more for scene: Pairwise consistency learning for scene classification. In *Advances in Neural Information Processing Systems*, pages 4987–4999. Curran Associates, Inc., 2021.
- [5] Gongwei Chen, Leyang Shen, Rui Shao, Xiang Deng, and Liqiang Nie. Lion: Empowering multimodal large language model with dual-level visual knowledge. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 26540–26550, 2024. 2
- [6] Gongwei Chen, Lirong Jie, Lexiao Zou, Weili Guan, Miao Zhang, and Liqiang Nie. Enhancing GUI agent with uncertainty-aware self-trained evaluator. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025. 2
- [7] Gongwei Chen, Xurui Zhou, Rui Shao, Yibo Lyu, Kaiwen Zhou, Shuai Wang, Wentao Li, Yinchuan Li, Zhonggang Qi, and Liqiang Nie. Less is more: Empowering gui agent with context-aware simplification. *arXiv preprint arXiv:2507.03730*, 2025. 2, 4
- [8] Jingxuan Chen, Derek Yuen, Bin Xie, Yuhao Yang, Gongwei Chen, Zhihao Wu, Li Yixing, Xurui Zhou, Weiwen Liu, Shuai Wang, et al. Spa-bench: A comprehensive benchmark for smartphone agent evaluation. In *NeurIPS 2024 Workshop on Open-World Agents*, 2024. 2
- [9] Liang Chen, Haozhe Zhao, Tianyu Liu, Shuai Bai, Junyang Lin, Chang Zhou, and Baobao Chang. An image is worth 1/2 tokens after layer 2: Plug-and-play inference acceleration for large vision-language models. In *European Conference on Computer Vision*, pages 19–35. Springer, 2024. 4
- [10] Xinyang Chen, Sinan Wang, Mingsheng Long, and Jianmin Wang. Transferability vs. discriminability: Batch spectral penalization for adversarial domain adaptation. In *International conference on machine learning*, pages 1081–1090. PMLR, 2019. 2
- [11] Kanzhi Cheng, Qiushi Sun, Yougang Chu, Fangzhi Xu, Yantao Li, 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*. arXiv, 2024. arXiv:2401.10935 [cs]. 1, 2
- [12] DeepSeek-AI, Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, Xiaokang Zhang, Xingkai Yu, Yu Wu, Z. F. Wu, Zhibin Gou, Zhihong Shao, Zhuoshu Li, Ziyi Gao, Aixin Liu, Bing Xue, Bingxuan Wang, Bochao Wu, Bei Feng, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, Damai Dai, Deli Chen, Dongjie Ji, Erhang Li, Fangyun Lin, Fucong Dai, Fuli Luo, Guangbo Hao, Guanting Chen, Guowei Li, H. Zhang, Han Bao, Hanwei Xu, Haocheng Wang, Honghui Ding, Huajian Xin, Huazuo Gao, Hui Qu, Hui Li, Jianzhong Guo, Jiashi Li, Jiawei Wang, Jingchang Chen, Jingyang Yuan, Junjie Qiu, Junlong Li, J. L. Cai, Jiaqi Ni, Jian Liang, Jin Chen, Kai Dong, Kai Hu, Kaige Gao, Kang Guan, Kexin Huang, Kuai Yu, Lean Wang, Lecong Zhang, Liang Zhao, Litong Wang, Liyue Zhang, Lei Xu, Leyi Xia, Mingchuan Zhang, Minghua Zhang, Minghui Tang, Meng Li, Miaojun Wang, Mingming Li, Ning Tian, Panpan Huang, Peng Zhang, Qiancheng Wang, Qinyu Chen, Qiushi Du, Ruiqi Ge, Ruisong Zhang, Ruizhe Pan, Runji Wang, R. J. Chen, R. L. Jin, Ruyi Chen, Shanghao Lu, Shangyan Zhou, Shanhuang Chen, Shengfeng Ye, Shiyu Wang, Shuiping Yu, Shunfeng Zhou, Shuting Pan, S. S. Li, Shuang Zhou, Shaoqing Wu, Shengfeng Ye, Tao Yun, Tian Pei, Tianyu Sun, T. Wang, Wangding Zeng, Wanxia Zhao, Wen Liu, Wenfeng Liang, Wenjun Gao, Wenqin Yu, Wentao Zhang, W. L. Xiao, Wei An, Xiaodong Liu, Xiaohan Wang, Xiaokang Chen, Xiaotao Nie, Xin Cheng, Xin Liu, Xin Xie, Xingchao Liu, Xinyu Yang, Xinyuan Li, Xuecheng Su, Xuheng Lin, X. Q. Li, Xiangyue Jin, Xiaojin Shen, Xiaosha Chen, Xiaowen Sun, Xiaoxiang Wang, Xinnan Song, Xinyi Zhou, Xianzu Wang, Xinxia Shan, Y. K. Li, Y. Q. Wang, Y. X. Wei, Yang Zhang, Yanhong Xu, Yao Li, Yao Zhao, Yaofeng Sun, Yao-hui Wang, Yi Yu, Yichao Zhang, Yifan Shi, Yiliang Xiong, Ying He, Yishi Piao, Yisong Wang, Yixuan Tan, Yiyang Ma, Yiyuan Liu, Yongqiang Guo, Yuan Ou, Yudian Wang, Yue Gong, Yuheng Zou, Yujia He, Yunfan Xiong, Yuxiang Luo, Yuxiang You, Yuxuan Liu, Yuyang Zhou, Y. X. Zhu, Yanhong Xu, Yanping Huang, Yaohui Li, Yi Zheng, Yuchen Zhu, Yunxian Ma, Ying Tang, Yukun Zha, Yuting Yan, Z. Z. Ren, Zehui Ren, Zhangli Sha, Zhe Fu, Zhean Xu, Zhenda Xie, Zhengyan Zhang, Zhewen Hao, Zhicheng Ma, Zhigang Yan, Zhiyu Wu, Zihui Gu, Zijia Zhu, Zijun Liu, Zilin Li, Ziwei Xie, Ziyang Song, Zizheng Pan, Zhen Huang, Zhipeng Xu, Zhongyu Zhang, and Zhen Zhang. DeepSeek-R1: Incentivizing Reasoning Capability in LLMs via Reinforcement Learning, 2025. arXiv:2501.12948 [cs]. 2
- [13] Longxi Gao, Li Zhang, and Mengwei Xu. Uishift: Enhancing vlm-based gui agents through self-supervised reinforcement learning. *arXiv preprint arXiv:2505.12493*, 2025. 6, 7
- [14] Boyu Gou, Ruohan Wang, Boyuan Zheng, Yanan Xie, Cheng Chang, Yiheng Shu, Huan Sun, and Yu Su. Navigating the digital world as humans do: Universal visual grounding for GUI agents. In *The Thirteenth International Conference on Learning Representations*, 2025. 1
- [15] 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. 2, 6, 7
- [16] Junpeng Jiang, Gangyi Hong, Miao Zhang, Hengtong Hu, Kun Zhan, Rui Shao, and Liqiang Nie. Dive: Efficient multi-view driving scenes generation based on video diffusion transformer. *arXiv preprint arXiv:2504.19614*, 2025. 2
- [17] Hanyu Lai, Xiao Liu, Iat Long Iong, Shuntian Yao, Yuxuan Chen, Pengbo Shen, Hao Yu, Hanchen Zhang, Xiaohan Zhang, Yuxiao Dong, and Jie Tang. AutoWebGLM: Bootstrap And Reinforce A Large Language Model-based Web Navigating Agent. In *Proceedings of the 30th ACM SIGKDD*

- Conference on Knowledge Discovery and Data Mining*. arXiv, 2024. arXiv:2404.03648 [cs]. 1
- [18] Wei Li, William E Bishop, Alice Li, Christopher Rawles, Folawiyi Campbell-Ajala, Divya Tyamagundlu, and Oriana Riva. On the effects of data scale on ui control agents. *Advances in Neural Information Processing Systems*, 37:92130–92154, 2024. 6, 1
- [19] Wei Li, Bing Hu, Rui Shao, Leyang Shen, and Liqiang Nie. Lion-fs: Fast & slow video-language thinker as online video assistant. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 3240–3251, 2025. 2
- [20] Wei Li, Renshan Zhang, Rui Shao, Zhijian Fang, Kaiwen Zhou, Zhuotao Tian, and Liqiang Nie. Semanticvla: Semantic-aligned sparsification and enhancement for efficient robotic manipulation. *arXiv preprint arXiv:2511.10518*, 2025.
- [21] Wei Li, Renshan Zhang, Rui Shao, Jie He, and Liqiang Nie. Cogvla: Cognition-aligned vision-language-action model via instruction-driven routing & sparsification. *arXiv preprint arXiv:2508.21046*, 2025.
- [22] Zaijing Li, Gongwei Chen, Rui Shao, Yuquan Xie, Dongmei Jiang, and Liqiang Nie. Enhancing emotional generation capability of large language models via emotional chain-of-thought. *arXiv preprint arXiv:2401.06836*, 2024.
- [23] Zaijing Li, Yuquan Xie, Rui Shao, Gongwei Chen, Dongmei Jiang, and Liqiang Nie. Optimus-1: Hybrid multimodal memory empowered agents excel in long-horizon tasks. In *Advances in Neural Information Processing Systems*, pages 49881–49913, 2024. 2
- [24] Zaijing Li, Yuquan Xie, Rui Shao, Gongwei Chen, Weili Guan, Dongmei Jiang, and Liqiang Nie. Optimus-3: Towards generalist multimodal minecraft agents with scalable task experts. *arXiv preprint arXiv:2506.10357*, 2025. 2
- [25] Zaijing Li, Yuquan Xie, Rui Shao, Gongwei Chen, Dongmei Jiang, and Liqiang Nie. Optimus-2: Multimodal minecraft agent with goal-observation-action conditioned policy. In *Proceedings of the computer vision and pattern recognition conference*, pages 9039–9049, 2025. 2
- [26] Shuquan Lian, Yuhang Wu, Jia Ma, Zihan Song, Bingqi Chen, Xiawu Zheng, and Hui Li. Ui-agile: Advancing gui agents with effective reinforcement learning and precise inference-time grounding. *arXiv preprint arXiv:2507.22025*, 2025. 1, 6, 7
- [27] Yuhang Liu, Pengxiang Li, Congkai Xie, Xavier Hu, Xiaotian Han, Shengyu Zhang, Hongxia Yang, and Fei Wu. Infigui-r1: Advancing multimodal gui agents from reactive actors to deliberative reasoners. *arXiv preprint arXiv:2504.14239*, 2025. 6, 7
- [28] Ziyu Liu, Zeyi Sun, Yuhang Zang, Xiaoyi Dong, Yuhang Cao, Haodong Duan, Dahua Lin, and Jiaqi Wang. Visual-RFT: Visual Reinforcement Fine-Tuning, 2025. arXiv:2503.01785 [cs]. 1
- [29] Quanfeng Lu, Wenqi Shao, Zitao Liu, Fanqing Meng, Boxuan Li, Botong Chen, Siyuan Huang, Kaipeng Zhang, Yu Qiao, and Ping Luo. GUI Odyssey: A Comprehensive Dataset for Cross-App GUI Navigation on Mobile Devices. *arXiv preprint*, 2024. arXiv:2406.08451 [cs]. 2, 6, 1
- [30] Zhengxi Lu, Yuxiang Chai, Yaxuan Guo, Xi Yin, Liang Liu, Hao Wang, Guanqing Xiong, and Hongsheng Li. UI-R1: Enhancing Action Prediction of GUI Agents by Reinforcement Learning, 2025. arXiv:2503.21620 [cs] version: 1. 1
- [31] Run Luo, Lu Wang, Wanwei He, and Xiaobo Xia. Gui-r1: A generalist r1-style vision-language action model for gui agents. *arXiv preprint arXiv:2504.10458*, 2025. 1, 2, 6, 7
- [32] Zhihao Luo, Wentao Yan and Jingyu Gong, Min Wang, Zhizhong Zhang, Xuhong Wang, Yuan Xie, and Xin Tan. Navimaster: Learning a unified policy for gui and embodied navigation tasks. *arXiv preprint arXiv:2508.02046*, 2025. 2
- [33] Trung Quoc Luong, Xinbo Zhang, Zhanming Jie, Peng Sun, Xiaoran Jin, and Hang Li. ReFT: Reasoning with Reinforced Fine-Tuning, 2024. arXiv:2401.08967 [cs]. 1
- [34] Yibo Lyu, Rui Shao, Gongwei Chen, Yijie Zhu, Weili Guan, and Liqiang Nie. Puma: Layer-pruned language model for efficient unified multimodal retrieval with modality-adaptive learning. In *Proceedings of the 33rd ACM International Conference on Multimedia*, 2025. 2
- [35] 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. 6
- [36] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. pages 1–12, 2017. arXiv: 1707.06347. 2
- [37] Rui Shao, Xiangyuan Lan, and Pong C Yuen. Deep convolutional dynamic texture learning with adaptive channel-discriminability for 3d mask face anti-spoofing. In *2017 IEEE International Joint Conference on Biometrics (IJCB)*, pages 748–755. IEEE, 2017. 2
- [38] Rui Shao, Xiangyuan Lan, Jiawei Li, and Pong C Yuen. Multi-adversarial discriminative deep domain generalization for face presentation attack detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10023–10031, 2019.
- [39] Rui Shao, Tianxing Wu, and Ziwei Liu. Detecting and grounding multi-modal media manipulation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6904–6913, 2023.
- [40] Rui Shao, Tianxing Wu, Jianlong Wu, Liqiang Nie, and Ziwei Liu. Detecting and grounding multi-modal media manipulation and beyond. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2024. 2
- [41] Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. DeepSeekMath: Pushing the Limits of Mathematical Reasoning in Open Language Models, 2024. arXiv:2402.03300 [cs]. 2
- [42] Leyang Shen, Gongwei Chen, Rui Shao, Weili Guan, and Liqiang Nie. Mome: Mixture of multimodal experts for generalist multimodal large language models. In *Advances in Neural Information Processing Systems*, pages 42048–42070, 2024. 2
- [43] Huajie Tan, Yuheng Ji, Xiaoshuai Hao, Minglan Lin, Pengwei Wang, Zhongyuan Wang, and Shanghang Zhang. Reason-RFT: Reinforcement Fine-Tuning for Visual Reasoning, 2025. arXiv:2503.20752 [cs]. 1

- [44] Junyang Wang, Haiyang Xu, Jiabo Ye, Ming Yan, Weizhou Shen, Ji Zhang, Fei Huang, and Jitao Sang. Mobile-Agent: Autonomous Multi-Modal Mobile Device Agent with Visual Perception. *arXiv preprint*, 2024. arXiv:2401.16158 [cs]. 2
- [45] Zhiyong Wu, Zhenyu Wu, Fangzhi Xu, Yian Wang, Qiushi Sun, Chengyou Jia, Kanzhi Cheng, Zichen Ding, Liheng Chen, Paul Pu Liang, and Yu Qiao. OS-ATLAS: A Foundation Action Model for Generalist GUI Agents. *arXiv preprint*, 2024. arXiv:2410.23218 [cs]. 1, 6, 7
- [46] Bin Xie, Rui Shao, Gongwei Chen, Kaiwen Zhou, Yinchuan Li, Jie Liu, Min Zhang, and Liqiang Nie. Gui-explorer: Autonomous exploration and mining of transition-aware knowledge for gui agent. In *Annual Meeting of the Association for Computational Linguistics (ACL)*, 2025.
- [47] Yuquan Xie, Zaijing Li, Rui Shao, Gongwei Chen, Kaiwen Zhou, Yinchuan Li, Dongmei Jiang, and Liqiang Nie. Mirage-1: Augmenting and updating gui agent with hierarchical multimodal skills. *arXiv preprint arXiv:2506.10387*, 2025. 2
- [48] Yiheng Xu, Zekun Wang, Junli Wang, Dunjie Lu, Tianbao Xie, Amrita Saha, Doyen Sahoo, Tao Yu, and Caiming Xiong. Aguis: Unified Pure Vision Agents for Autonomous GUI Interaction. *arXiv preprint*, 2024. arXiv:2412.04454 [cs]. 1
- [49] Danyang Zhang, Situo Zhang, Ziyue Yang, Zichen Zhu, Zihan Zhao, Ruisheng Cao, Lu Chen, and Kai Yu. Progrm: Build better gui agents with progress rewards. *arXiv preprint arXiv:2505.18121*, 2025. 3
- [50] Jiwen Zhang, Yaqi Yu, Minghui Liao, Wentao Li, Jihao Wu, and Zhongyu Wei. Ui-hawk: Unleashing the screen stream understanding for gui agents. *Preprints*, 2024. 1
- [51] Jiwen Zhang, Yaqi Yu, Minghui Liao, Wentao Li, Jihao Wu, and Zhongyu Wei. UI-Hawk: Unleashing the Screen Stream Understanding for GUI Agents, 2024. 2
- [52] Renshan Zhang, Rui Shao, Gongwei Chen, Miao Zhang, Kaiwen Zhou, Weili Guan, and Liqiang Nie. Falcon: Resolving visual redundancy and fragmentation in high-resolution multimodal large language models via visual registers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2025. 2
- [53] Zhi Zhang, Srishti Yadav, Fengze Han, and Ekaterina Shutova. Cross-modal information flow in multimodal large language models. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 19781–19791, 2025. 4
- [54] Yijie Zhu, Rui Shao, Ziyang Liu, Jie He, Jizhihui Liu, Jiuru Wang, and Zitong Yu. H-gar: A hierarchical interaction framework via goal-driven observation-action refinement for robotic manipulation. *arXiv preprint arXiv:2511.17079*, 2025. 2
- [55] Lexiao Zou, Gongwei Chen, Yanda Chen, and Miao Zhang. Enhancing diffusion-based dataset distillation via adversary-guided curriculum sampling. In *2025 IEEE International Conference on Multimedia and Expo (ICME)*, pages 1–6. IEEE, 2025. 2