Mobile-Agent-V: A Video-Guided Approach for Effortless and Efficient Operational Knowledge Injection in Mobile Automation

Anonymous ACL submission

Abstract

The exponential rise in mobile device usage necessitates streamlined automation for effective task management, yet many AI frameworks fall short due to inadequate operational expertise. While manually written knowledge can bridge this gap, it is often burdensome and inefficient. We introduce Mobile-Agent-V, an innovative framework that utilizes video as a guiding tool to effortlessly and efficiently inject operational knowledge into mobile automation processes. By deriving knowledge directly from video content, Mobile-Agent-V eliminates manual intervention, significantly reducing the effort and time required for knowledge acquisition. To rigorously evaluate this approach, we propose Mobile-Knowledge, a benchmark tailored to assess the impact of external knowledge on mobile agent performance. Our experimental findings demonstrate that Mobile-Agent-V enhances performance by 36% compared to existing methods, underscoring its effortless and efficient advantages in mobile automation.

1 Introduction

The reliance on mobile devices has increased, with users performing numerous operations daily, underscoring the need for streamlined interactions. Currently, the development of Multimodal Large Language Models (MLLMs) has notably improved mobile device operating frameworks, using these models as intelligent agents (Liu et al., 2023b; Zhu et al., 2023; Ye et al., 2023a; Dai et al., 2023; Liu et al., 2023a; Chen et al., 2023; Bai et al., 2023; Ye et al., 2023b; Wang et al., 2023; Lu et al., 2024a; Ye et al., 2024; Wu et al., 2024; Qin et al., 2025). These frameworks leverage agents' perception, decision-making, and reflection to perform complex tasks across multiple applications, thereby broadening mobile devices' autonomous capabilities.

Despite progress, existing approaches remain constrained by limited operational knowledge. As shown in Figure 1(a), agents struggle to complete
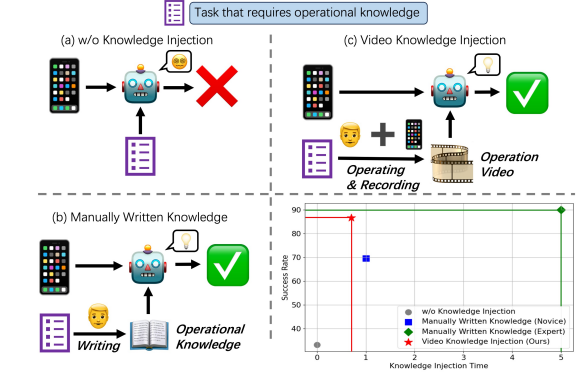

Figure 1: (a) Mobile agents often struggle to complete tasks due to a lack of knowledge. (b) Manually written knowledge requires a high level of human expertise and precision, leading to significant differences in performance depending on whether novices or experts author the content. (c) Mobile-Agent-V learns directly from video, bypassing the need for human expertise. It is more efficient and can even exceed the effectiveness of manually written knowledge. In the evaluation of Mobile-Knowledge, Mobile-Agent-V achieves performance comparable to human experts while saving over 80% of the time required for knowledge injection.

certain tasks when lacking operational knowledge. This is primarily due to the inadequacy of training data to encompass all scenarios. Additionally, the unique nature of some scenarios prevents existing agent knowledge from generalizing effectively. To address this issue, current frameworks typically incorporate manually written knowledge into the agent framework, delivered in textual form (Yang et al., 2023; Li et al., 2024b; Wang et al., 2024c,b; Agashe et al., 2025). However, as depicted in Figure 1(b), this approach is highly sensitive to the quality of human expertise. In order to achieve better outcomes, the involvement of experts becomes necessary. This reliance on manually authored knowledge increases the cost of knowledge injection and reduces efficiency.

To develop methods of knowledge injection that are less reliant on human quality and more effi-

cient, we aim to use knowledge sources in their natural, unprocessed forms. Observations of existing work have shown that video can enhance effectiveness, inspiring us to extract procedural knowledge directly from instructional videos (Wang et al., 2024e,a; Zhang et al., 2024c; Chane-Sane et al., 2023). These videos require users to perform and document an entire operation just once, which removes the need for further human involvement as in Figure 1(c). However, the frequent scene changes and high information density in instructional videos present significant challenges. Additionally, current large-scale visual models often have difficulty processing video input, hindering the ability of existing frameworks to effectively utilize video-based learning.

To address this, we introduce Mobile-Agent-V, a multi-agent framework that processes operational video inputs, extracts actionable knowledge, and applies it to mobile device interactions. To reduce keyframe redundancy while retaining crucial information, we use a sliding window mechanism, feeding a subset of keyframes into the decision agent. The video agent assesses the device's state and adaptively shifts the window forward, ensuring frames remain relevant for decision-making. Despite this, multi-frame inputs challenge MLLMs in maintaining contextual coherence. To enhance accuracy, we employ a reflection agent with long-chain-of-thought reasoning to analyze the video, refine decision outputs.

Existing mobile benchmarks predominantly assess a range of integrated capabilities—such as localization, planning, decision-making, which can conflict with evaluating knowledge utilization, making it difficult to evaluate the effect of knowledge injection alone. To address this, we introduce Mobile-Knowledge, a benchmark designed to specifically assess knowledge utilization efficacy. Utilizing straightforward tasks, it minimizes factors unrelated to knowledge injection. Experimental results indicate Mobile-Agent-V improves performance by 36% over existing frameworks, demonstrating its superiority in knowledge utilization.

Our summarized contributions are as follows:

- We introduce Mobile-Agent-V, a novel framework that applies video guidance to achieve effortless and efficient knowledge injection. Knowledge injection can be accomplished simply by performing the task once and recording a video, eliminating the need for

high-quality manual labor and lengthy knowledge construction time.

- We propose a multi-agent collaboration strategy to effectively extract and utilize knowledge from videos. To address the challenges of processing long-context video input, we introduce a sliding window strategy in conjunction with a video agent. By incorporating a deep-reflection agent, we further enhance decision accuracy.

- To focus on evaluating the effectiveness of knowledge utilization, we introduce Mobile-Knowledge, which comprises tasks that require procedural knowledge but demand minimal basic operational abilities. Experimental results demonstrate that Mobile-Agent-V achieves a 36% performance improvement over existing frameworks.

2 Related Work

2.1 GUI Agent

Intelligent agent frameworks using Large Language Models (LLMs) are advancing in GUI operations to enhance user experience (Wang et al., 2024d; Liu et al., 2025). HTML-based parsing is common on the Web due to its interpretability, while frameworks such as ChatGPT's assistant use visual perception (Zhou et al., 2023; Deng et al., 2023; Zheng et al., 2024; He et al., 2024; Lù et al., 2024; Yoran et al., 2024; Reddy et al., 2024). PC-based frameworks rely on system APIs for greater control (Zhang et al., 2024a; Tan et al., 2024; Xie et al., 2024). Mobile automation challenges involve providing agents with operational knowledge, which LLMs often lack. Existing approaches often involve costly training on operational data (Hong et al., 2023; Cheng et al., 2024; You et al., 2024; Zhang et al., 2024b; Chen and Li, 2024; Lu et al., 2024b; Chai et al., 2024; Rawles et al., 2024; Xu et al., 2024; Li et al., 2024a; Wan et al., 2024; Xing et al., 2024; Liu et al., 2024), extensive exploration (Yang et al., 2023; Wang et al., 2024c; Li et al., 2024b; Wang et al., 2025), or inefficiencies through manual knowledge (Wang et al., 2024b).

2.2 Video-guided Agent

Video guidance is crucial for training intelligent agents to effectively interact with dynamic environments. Initial efforts using large language models
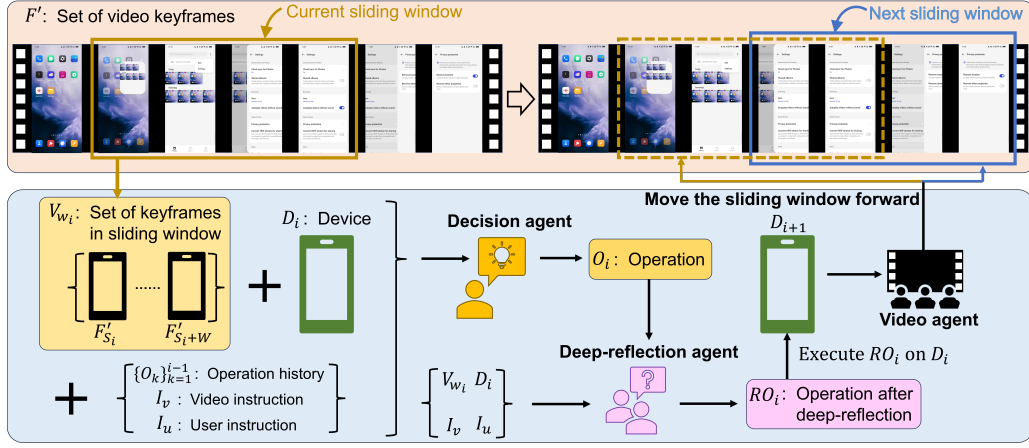
Figure 2: The framework of Mobile-Agent-V.

(LLMs) focused on video comprehension (Wang et al., 2024e). Beyond comprehension, video applications include automated video editing (Wang et al., 2024a), efficient frame retrieval (Zhang et al., 2024c), and robotics training via human demonstration videos (Chane-Sane et al., 2023). These practical uses showcase the expanding role of video-guided agents in various fields.

3 Mobile-Agent-V

This section introduces Mobile-Agent-V, a framework that enhances mobile automation through video guidance. We outline its key components, including video processing, sliding window, video agent, deep-reflection agent, decision agent, and explain how they work together to improve operational efficiency and accuracy.

3.1 Framework

The overall workflow of Mobile-Agent-V is shown in Figure 2. Given an input video V that captures a demonstrated task, the system first extracts keyframes F' through uniform sampling and redundancy removal. The execution begins with an initial sliding window positioned at the start of the keyframe sequence. At each iteration, the decision agent generates an action O_i based on the current window, video instructions, and historical decisions. If the task is successfully completed, the process terminates. Otherwise, the deep-reflection agent validates and refines the action to ensure alignment with the demonstrated task. The refined decision RO_i is then executed on the device, updating its state to D_{i+1}. The video agent subsequently determines the next window starting point S_{i+1}, facilitating a dynamic adjustment of the observation scope as the task progresses. This iterative proce-

dure continues until the task is completed or the predefined maximum exploration limit is reached. The complete pipeline is outlined in Algorithm 1.

3.2 Video Processing

Traditional uniform sampling suits real-world videos with static scenes and smooth motion. However, in mobile recordings, most frames are static, while rapid changes occur due to human interaction and fast device responses, rendering uniform sampling ineffective for mobile videos. To address this, we first uniformly sample the V at a frequency d to obtain the keyframe set F:

$$F = \text{Uniform\_Sampling}(V, d) \qquad (1)$$

Next, we compute the similarity between consecutive keyframes and remove those with similarity above a threshold s, resulting in a reduced set F_s:

$$F_s = \{f_i \in F \mid \text{sim}(f_i, f_{i+1}) \leq s\} \qquad (2)$$

Finally, we filter out keyframes with temporal gaps smaller than a threshold f_s, yielding the final set of keyframes F':

$$F' = \{f_i \in F_s \mid t_{i+1} - t_i \geq d\} \qquad (3)$$

where t_i represents the frame index of f_i.

3.3 Sliding Window

To improve video comprehension by MLLMs, we reduce the input length by selecting only the keyframes relevant to the current operation. This is achieved using a sliding window, where the keyframes between the window's start and end points V_w serve as the input for decision-making:

$$V_w = \{F'_k\}_{k=S_i}^{S_i+W} \qquad (4)$$

where the w is the length of the window.

3

3.4 Decision Agent

Action Space. The decision agent is responsible for generating actions that alter the device state. Mobile-Agent-V defines six fundamental actions: *Click*, *Scroll*, *Type*, *Back*, *Home*, and *Done*. A detailed description of the operating space is shown in the Appendix A.1.6.

Decision Making. Unlike prior methods that rely on internal operational knowledge, the decision agent in Mobile-Agent-V derives actions directly from video content. This imposes higher demands on contextual adherence. By leveraging the sliding window mechanism, we filter out irrelevant frames, reducing input length while preserving critical information. The i-th operation O_i follows the steps outlined in the following equation:

$$O_i = Da(Vw_i, I_v, D_i, I_u, \{O_k\}_{k=1}^{i-1}) \quad (5)$$

where $Da(\cdot)$ is the decision agent, I_v is the instruction completed in the video, D_i is the screenshot of the device during the i-th operation, and I_u is the instruction that the user will complete on the current device. Besides this, to track the progress, we also provide the historical operations $\{O_k\}_{k=1}^{i-1}$ to the decision agent.

3.5 Deep-Reflection Agent

Even with a sliding window, low-quality keyframes require larger window sizes because a smaller window may be filled with redundant frames, excluding important keyframes. In cases where perfect keyframe extraction is not possible, the decision agent struggles with long multi-frame sequences. To overcome this, we introduce the deep-reflection agent, which validates and refines the decision agent's outputs. It systematically analyzes each operation in the video, identifies the current device state, checks if the decision agent's action matches the corresponding video operation, and refines the action based on the trajectory if discrepancies are found. This reflection mechanism enhances decision accuracy by ensuring strict adherence to the demonstrated operations, leading to a final refined decision RO_i, formulated as follows:

$$RO_i = Ra(Vw_i, I_v, D_i, I_u, O_i) \quad (6)$$

3.6 Video Agent

To dynamically adjust the sliding window throughout task execution, we introduce the video agent. Initially, the window spans from the first keyframe

---

Algorithm 1 Mobile-Agent-V pipeline

Input: Video V, Window length W, Video task I_v, User task I_u, Decision agent Da, Reflection agent Ra, Video agent Va, Max explorations M_e

1: **Initialization:**
2: Obtain F' from V as Equ. (1) (2) (3)
3: $S_1 \leftarrow 1$
4: **for** $i = 1$ **to** M_e **do**
5:     Obtain V_{w_i} from F'_k as Equ.(4)
6:     $O_i \leftarrow Da(Vw_i, I_v, D_i, I_u, \{O_k\}_{k=1}^{i-1})$
7:     **if** $O_i ==$ Done **then**
8:         break
9:     **end if**
10:     $RO_i \leftarrow Ra(Vw_i, I_v, D_i, I_u, O_i)$
11:     $D_{i+1} \leftarrow$ Execute RO_i on Device
12:     $R_i \leftarrow \{D_k\}_{k=i}^{i+1}$
13:     $S_{i+1} \leftarrow Va(Vw_i, I_v, R_i, I_u)$
14: **end for**

---

to the W-th keyframe. After each operation, the video agent analyzes the screenshots before and after the operation, keyframes within the current window, and user inputs to identify the corresponding keyframe. Then, it determines the updated window starting point, ensuring adaptive progression. The following is the formula for obtaining the starting point of the $i + 1$-th sliding window:

$$S_{i+1} = Va(Vw_i, I_v, R_i, I_u) \quad (7)$$

where $Va(\cdot)$ is the video agent, and R_i is the set of screenshots before and after the operation:

$$R_i = \{D_k\}_{k=i}^{i+1} \quad (8)$$

4 Experiments

This section presents a comprehensive evaluation of Mobile-Agent-V. We first introduce the evaluation methodology. Next, we describe the experimental setup. We then report the main results. Finally, we conduct qualitative analyses and ablation studies to further examine the contributions of individual components.

4.1 Evaluation

In this subsection, we will introduce the evaluation benchmarks and corresponding metrics.

4.1.1 Benchmark

Mobile-Knowledge. Traditional benchmarks like AITW assess agents' planning and operational

skills, including task decomposition, UI element localization, and gesture execution. While these metrics are effective for evaluating basic competencies, they often mix inherent abilities with external knowledge integration. Mobile-Knowledge specifically targets the second dimension. This benchmark minimizes planning and operational complexity, instead emphasizing tasks reliant on knowledge not covered in standard agent training data. We crafted 30 device-specific tasks, categorized as basic, normal, and advanced instructions, each requiring increasing levels of specialized knowledge. Each instruction provides clear directives to avoid biases not related to knowledge integration. For each task, corresponding videos and manually compiled knowledge were provided, with professional annotators supplying the expertise-driven knowledge. Details of the tasks are available in Appendix A.3.1.

AndroidWorld-Knowledge. To evaluate the knowledge generalizability, we developed AndroidWorld-Knowledge within the Android World (Rawles et al., 2024) environment. We selected five applications—Expense, Marker, Receipt, SportsTracker, and Tasks—comprising a total of 48 tasks that demand substantial operational knowledge. Within each scenario, only the operation video and manually authored knowledge for the simplest task were provided. This means other tasks in the scenario lacked direct video guidance, relying instead on the least complex task video as a reference. This design assesses the framework's ability to generalize knowledge application beyond direct video instructions. Details of the tasks are available in Appendix A.3.2.

4.1.2 Metrics

We evaluate Mobile-Agent-V and other baselines on Mobile-Knowledge using four key metrics: Success Rate (SR), Completion Rate (CR), Decision Accuracy (DA), and Step Count (Step). The detailed explanation of the evaluation metrics is presented in the Appendix A.3.3. For AndroidWorld-Knowledge, we follow existing studies by employing SR as a metric to evaluate performance.

4.2 Setup

Baselines. We compare Mobile-Agent-V with several open-source agent frameworks, including AppAgent-v1 (Yang et al., 2023), AppAgent-v2 (Li et al., 2024b), Mobile-Agent-v1 (Wang et al.,

2024c), Mobile-Agent-v2 (Wang et al., 2024b) and Agent-S2 (Agashe et al., 2025). For baselines, we utilize manually written knowledge provided by the benchmark for knowledge injection.

Models. Both Mobile-Agent-V and baselines utilize GPT-4o as their base model. The model is accessed via the official API with default hyperparameters.

Device and Interaction. Experiments on Mobile-Knowledge are conducted on a OnePlus 7 Pro smartphone using the Android Debug Bridge (ADB) for interaction.

4.3 Main Results

In this subsection, we will analyze the performance of different methods on the Mobile-Knowledge and AndroidWorld-Knowledge benchmarks.

4.3.1 Mobile-Knowledge

The results on the Mobile-Knowledge benchmark highlight the effectiveness of Mobile-Agent-V, which utilizes operation video for knowledge injection. Compared to baseline methods that rely on manually written knowledge, Mobile-Agent-V shows a significant improvement in metrics such as SR, CR, and DA, with enhancements of up to 23.4% over the best-performing baseline. Additionally, Mobile-Agent-V achieves greater action efficiency, as evidenced by a reduction in the Step metric. These outcomes underscore the advantages of integrating operation videos, offering a more dynamic and comprehensive understanding of tasks than static instructional text.

4.3.2 AndroidWorld-Knowledge

On the AndroidWorld-Knowledge benchmark, Mobile-Agent-V demonstrates a substantial improvement in SR over baselines, achieving a 31.3% SR. This represents a significant increase of at least 12.4% compared to the best baseline, highlighting the effectiveness of utilizing operation videos for knowledge integration. The notable performance gain emphasizes Mobile-Agent-V's capability to enhance generalizability and operational efficiency in diverse GUI tasks, surpassing traditional approaches that depend solely on manually written instructions. Since AndroidWorld-Knowledge provides only one video per scenario, it facilitates the evaluation of generalization when discrepancies arise between the operation video and the actual task. We will conduct a detailed analysis of the generalization derived from video knowledge in

5

Method	Knowledge Injection	SR	CR	DA	Step
AppAgent-v1 (Yang et al., 2023)	Manually Written	46.7	52.5	43.6	12.2
AppAgent-v2 (Li et al., 2024b)	Manually Written	60.0	67.3	57.7	10.8
Mobile-Agent-v1 (Wang et al., 2024c)	Manually Written	43.4	51.3	41.0	12.2
Mobile-Agent-v2 (Wang et al., 2024b)	Manually Written	56.6	59.8	54.8	11.4
Agent-S2 (Agashe et al., 2025)	Manually Written	63.3	73.9	60.1	13.6
Mobile-Agent-V (Ours)	Operation Video	**86.7**	**93.4**	**79.4**	**7.3**

Table 1: Evaluation results on Mobile-Knowledge benchmark.

Method	SR
AppAgent-v1 (Yang et al., 2023)	14.6
AppAgent-v2 (Li et al., 2024b)	18.9
Mobile-Agent-v1 (Wang et al., 2024c)	12.5
Mobile-Agent-v2 (Wang et al., 2024b)	16.7
Agent-S2 (Agashe et al., 2025)	18.9
Mobile-Agent-V (Ours)	**31.3**

Table 2: Evaluation results on AndroidWorld-Knowledge benchmark.

Section 4.4.1.

4.4 Analysis

We conducted analytical experiments on the framework's configuration using the Mobile-Knowledge.

4.4.1 Generalization from Videos

The Video-Misaligned task modifies original instructions so the video's operational logic aligns with the user task, but actions differ. This tests Mobile-Agent-V's ability to generalize from video demonstrations. As shown in Figure 3, Mobile-Agent-V's performance drops under Video-Misaligned conditions; basic instructions stay stable, while normal and advanced ones decline in SR and DA. Yet, the system still completes tasks competently, indicating its ability to generalize beyond direct instruction mapping. These results emphasize the importance of diverse video demonstrations for enhancing cross-instruction generalization.

Mobile-Agent-V's ability to generalize from videos is a key strength demonstrated on the AndroidWorld-Knowledge benchmark. In this benchmark, we provided only a single video or manually written knowledge for the simplest task in each of the five scenarios. As shown in Table 2, despite the potential discrepancies between the provided videos and the actual tasks, Mobile-Agent-V achieved a SR of 31.3%, significantly outperforming baselines. This indicates that Mobile-Agent-V

can effectively extrapolate from limited video input, generalizing to more complex tasks without direct video guidance. This capability underscores the adaptability and robustness of our video-guided approach, which is essential for practical mobile automation applications where task-specific video resources may be limited or unavailable.

4.4.2 Impact of Window Size

Figure 4 illustrates the effect of window size on task performance. Larger windows generally improve SR, CR, and DA while reducing steps, particularly for more complex tasks. However, beyond a certain threshold, further increasing the window size yields diminishing returns, with some metrics even declining. This decline is likely due to the introduction of irrelevant information, which interferes with decision-making. These findings highlight the importance of balancing temporal context to maximize efficiency.

4.4.3 Impact of Keyframe Quality

To investigate the impact of keyframe quality, we compare artificial sampling, where keyframes are manually selected to avoid redundancy and omission, with our uniform sampling and filtering strategy in Figure 5. As expected, manually chosen keyframes yield slightly better results, confirming that high-quality keyframes enhance performance. However, the gap between our method and manual selection remains small, demonstrating the effectiveness of our method in preserving essential task-relevant information.

4.4.4 Impact of Knowledge Injection Method

Figure 3 highlights the considerable impact of the knowledge injection method on performance and efficiency. Mobile-Agent-V utilizes operation videos, achieving a high SR of 86.7% while reducing knowledge injection time to just 0.7 minutes on average. It balances the benefits of novice and expert-level manually written knowledge, which,
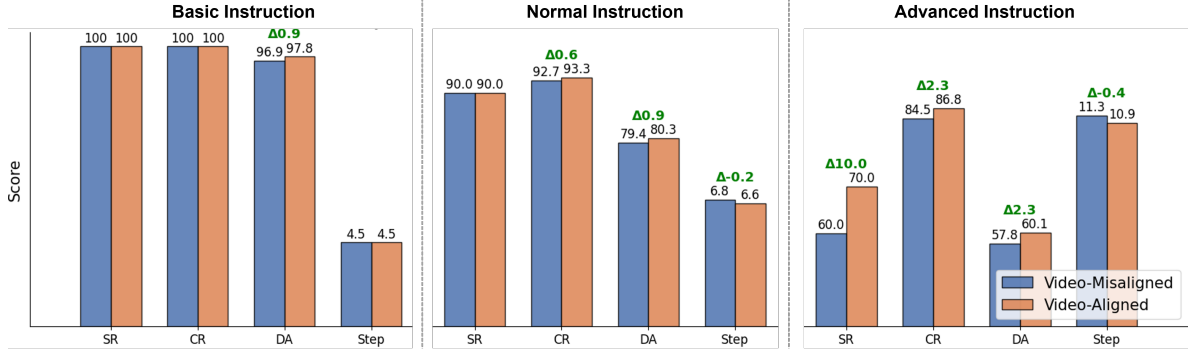
6

Figure 3: Comparison of video-misaligned instructions and video-aligned instructions. The video-aligned means that the video instruction is consistent with the user instruction, and the video-misaligned instruction is inconsistent.
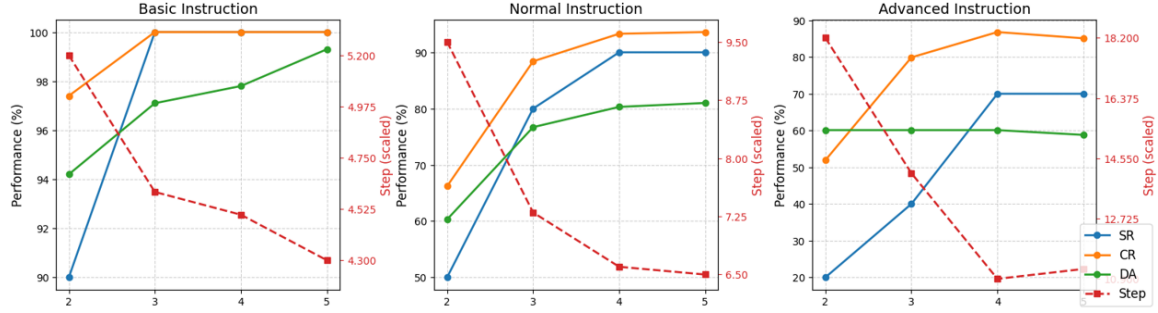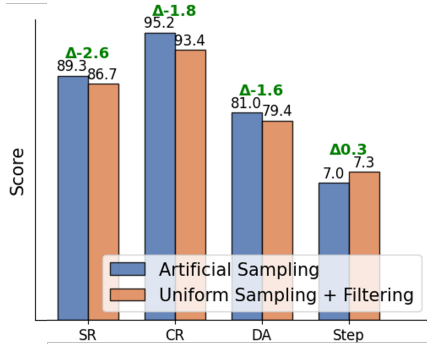

Figure 4: Comparison of different sliding window sizes.

Figure 5: Comparison of different keyframe quality.

Knowledge Injection Method	SR	Avg. Time
-	33.3	-
Manually Written - Novice	70.0	1 min
Manually Written - Expert	90.0	5 mins
Operation Video	86.7	0.7 min

Table 3: A comparison of the knowledge injection time and performance between video and manually written knowledge across varying levels of human expertise.

despite higher SRs, require substantial time—up to five minutes for expert knowledge. The efficiency of video-based knowledge aligns with Mobile-Agent-V's goals, focusing on seamless, efficient integration in mobile automation. Mobile-Agent-V provides an optimal solution, enhancing accessibility without sacrificing performance and avoiding the resource-intensive process of manual expertise.

4.5 Ablation Study

To evaluate the deep-reflection agent's effectiveness, we conducted an ablation study comparing its performance with and without the agent, as depicted in Figure 6. Results show that the deep-reflection agent consistently enhances decision-making across metrics. When SR and CR are high,

improvements are minor due to fewer errors by the decision agent. However, for complex tasks with lower baseline performance, the deep-reflection agent significantly boosts DA, refining actions and reducing inconsistencies in extended multi-frame reasoning. The Step metric shows slight changes, suggesting improved precision without major impacts on action efficiency. By correcting misalignments between predicted and actual actions, the agent mitigates cascading errors in long-horizon tasks, reduces reliance on perfect keyframe extraction, and enhances robustness and reliability in challenging visual conditions.

4.6 Case Study

Figure 7 presents a multi-agent collaboration scenario within Mobile-Agent-V. The decision agent analyzes keyframes from a sliding window to determine the operation but mistakenly skips the "con-
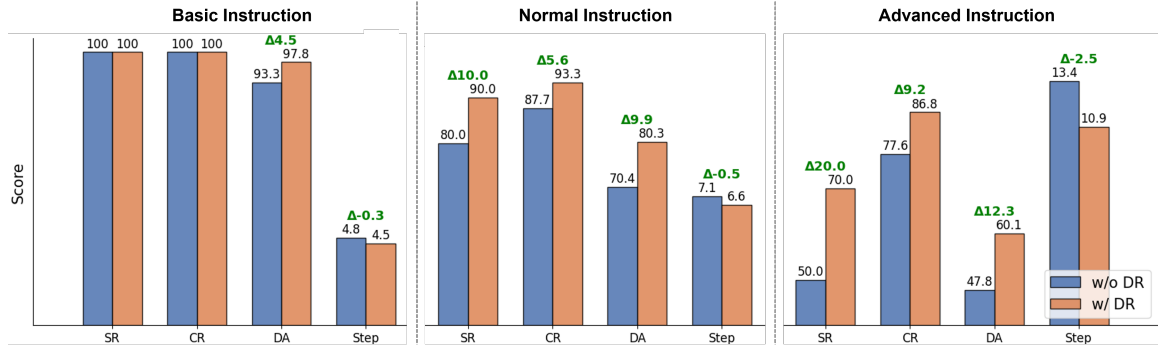
7

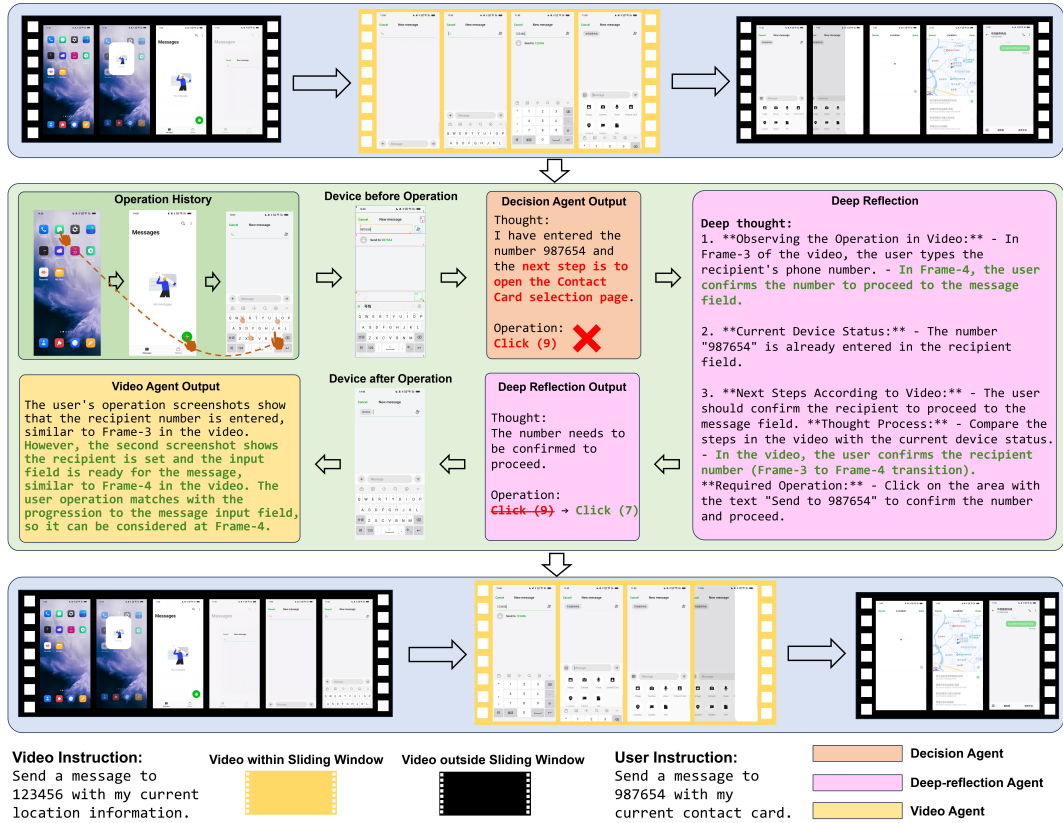Figure 6: Comparison of w/o DR and w/ DR across different instructions.

Figure 7: A complete execution case of Mobile-Agent-V. The decision agent initially makes an incorrect action, but the deep-reflection agent verifies the operation video, compares the device state, and corrects the action.

firm contact" step, highlighting multi-image action tracking challenges. The deep-reflection agent corrects this by identifying the misalignment and refining the decision to ensure accurate device operation. Meanwhile, the video agent anchors the device state to the fourth frame, then advances the window by two frames, allowing the system to accurately display the next interaction with the contact card.

5 Conclusion

We present Mobile-Agent-V, a video-guided framework that advances mobile automation by integrating dynamic, cost-effective operational knowledge. Using a sliding window mechanism, the video agent optimally selects keyframes, while the deep-reflection agent enhances decision accuracy through iterative reasoning. Experiments indicate Mobile-Agent-V's superior performance, with a 23.4% Success Rate improvement on Mobile-Knowledge and 12.4% on AndroidWorld-Knowledge. Mobile-Agent-V rivals expert-level written knowledge, reducing injection time by 86%, underscoring its potential for scalable learning. Mobile-Agent-V effectively transforms videos into operational knowledge, offering a streamlined path for agent development.

6 Limitations

While our method offers significant advantages, there are certain limitations to consider. Firstly, the dependency on video inputs may introduce variability in data quality; suboptimal recordings could impact the accuracy of knowledge extraction. Although the sliding window mechanism significantly enhances processing efficiency, there remains a possibility that essential frames could be overlooked during complex interactions. Furthermore, while our framework successfully generalizes across diverse tasks, its performance is somewhat contingent on the range and quality of video demonstrations available. Future work could focus on developing adaptive mechanisms to further improve both the efficiency and robustness of the system, ensuring it can handle a wider array of scenarios with varying video quality.

References

Saaket Agashe, Kyle Wong, Vincent Tu, Jiachen Yang, Ang Li, and Xin Eric Wang. 2025. Agent s2: A compositional generalist-specialist framework for computer use agents. *arXiv preprint arXiv:2504.00906*.

Jinze Bai, Shuai Bai, Shusheng Yang, Shijie Wang, Sinan Tan, Peng Wang, Junyang Lin, Chang Zhou, and Jingren Zhou. 2023. Qwen-vl: A frontier large vision-language model with versatile abilities. *arXiv preprint arXiv:2308.12966*.

Yuxiang Chai, Siyuan Huang, Yazhe Niu, Han Xiao, Liang Liu, Dingyu Zhang, Peng Gao, Shuai Ren, and Hongsheng Li. 2024. Amex: Android multi-annotation expo dataset for mobile gui agents. *arXiv preprint arXiv:2407.17490*.

Elliot Chane-Sane, Cordelia Schmid, and Ivan Laptev. 2023. Learning video-conditioned policies for unseen manipulation tasks. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 909–916. IEEE.

Jun Chen, Deyao Zhu Xiaoqian Shen Xiang Li, Zechun Liu Pengchuan Zhang, Raghuraman Krishnamoorthi Vikas Chandra Yunyang Xiong, and Mohamed Elhoseiny. 2023. Minigpt-v2: Large language model as a unified interface for vision-language multi-task learning. *arXiv preprint arXiv:2310.09478*.

Wei Chen and Zhiyuan Li. 2024. Octopus v2: On-device language model for super agent. *arXiv preprint arXiv:2404.01744*.

Kanzhi Cheng, Qiushi Sun, Yougang Chu, Fangzhi Xu, Yantao Li, Jianbing Zhang, and Zhiyong Wu. 2024. Seeclick: Harnessing gui grounding for advanced visual gui agents. *arXiv preprint arXiv:2401.10935*.

Wenliang Dai, Junnan Li, Dongxu Li, Anthony Meng Huat Tiong, Junqi Zhao, Weisheng Wang, Boyang Li, Pascale Fung, and Steven Hoi. 2023. Instructblip: Towards general-purpose vision-language models with instruction tuning. *arXiv preprint arXiv:2305.06500*.

Xiang Deng, Yu Gu, Boyuan Zheng, Shijie Chen, Samuel Stevens, Boshi Wang, Huan Sun, and Yu Su. 2023. Mind2web: Towards a generalist agent for the web. In *Thirty-seventh Conference on Neural Information Processing Systems*.

Hongliang He, Wenlin Yao, Kaixin Ma, Wenhao Yu, Yong Dai, Hongming Zhang, Zhenzhong Lan, and Dong Yu. 2024. Webvoyager: Building an end-to-end web agent with large multimodal models. *arXiv preprint arXiv:2401.13919*.

Wenyi Hong, Weihan Wang, Qingsong Lv, Jiazheng Xu, Wenmeng Yu, Junhui Ji, Yan Wang, Zihan Wang, Yuxiao Dong, Ming Ding, and Jie Tang. 2023. Cogagent: A visual language model for gui agents. *Preprint*, arXiv:2312.08914.

Wei Li, William E Bishop, Alice Li, Christopher Rawles, Folawiyo Campbell-Ajala, Divya Tyamagundlu, and Oriana Riva. 2024a. On the effects of data scale on ui control agents. In *The Thirty-eight Conference on Neural Information Processing Systems Datasets and Benchmarks Track*.

Yanda Li, Chi Zhang, Wanqi Yang, Bin Fu, Pei Cheng, Xin Chen, Ling Chen, and Yunchao Wei. 2024b. Appagent v2: Advanced agent for flexible mobile interactions. *arXiv preprint arXiv:2408.11824*.

Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. 2023a. Improved baselines with visual instruction tuning. *arXiv preprint arXiv:2310.03744*.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. 2023b. Visual instruction tuning. *arXiv preprint arXiv:2304.08485*.

William Liu, Liang Liu, Yaxuan Guo, Han Xiao, Weifeng Lin, Yuxiang Chai, Shuai Ren, Xiaoyu Liang, Linghao Li, Wenhao Wang, and 1 others. 2025. Llm-powered gui agents in phone automation: Surveying progress and prospects.

Xiao Liu, Bo Qin, Dongzhu Liang, Guang Dong, Hanyu Lai, Hanchen Zhang, Hanlin Zhao, Iat Long Iong, Jiadai Sun, Jiaqi Wang, and 1 others. 2024. Autoglm: Autonomous foundation agents for guis. *arXiv preprint arXiv:2411.00820*.

Haoyu Lu, Wen Liu, Bo Zhang, Bingxuan Wang, Kai Dong, Bo Liu, Jingxiang Sun, Tongzheng Ren, Zhuoshu Li, Hao Yang, and 1 others. 2024a. Deepseek-vl: towards real-world vision-language understanding. *arXiv preprint arXiv:2403.05525*.

Quanfeng Lu, Wenqi Shao, Zitao Liu, Fanqing Meng, Boxuan Li, Botong Chen, Siyuan Huang, Kaipeng Zhang, Yu Qiao, and Ping Luo. 2024b. Gui odyssey:

A comprehensive dataset for cross-app gui navigation on mobile devices. *arXiv preprint arXiv:2406.08451*.

Xing Han Lù, Zdeněk Kasner, and Siva Reddy. 2024. Weblinx: Real-world website navigation with multi-turn dialogue. *arXiv preprint arXiv:2402.05930*.

Yujia Qin, Yining Ye, Junjie Fang, Haoming Wang, Shihao Liang, Shizuo Tian, Junda Zhang, Jiahao Li, Yunxin Li, Shijue Huang, and 1 others. 2025. Ui-tars: Pioneering automated gui interaction with native agents. *arXiv preprint arXiv:2501.12326*.

Christopher Rawles, Sarah Clinckemaillie, Yifan Chang, Jonathan Waltz, Gabrielle Lau, Marybeth Fair, Alice Li, William Bishop, Wei Li, Folawiyo Campbell-Ajala, and 1 others. 2024. Androidworld: A dynamic benchmarking environment for autonomous agents. *arXiv preprint arXiv:2405.14573*.

Revanth Gangi Reddy, Sagnik Mukherjee, Jeonghwan Kim, Zhenhailong Wang, Dilek Hakkani-Tur, and Heng Ji. 2024. Infogent: An agent-based framework for web information aggregation. *arXiv preprint arXiv:2410.19054*.

Weihao Tan, Ziluo Ding, Wentao Zhang, Boyu Li, Bo-han Zhou, Junpeng Yue, Haochong Xia, Jiechuan Jiang, Longtao Zheng, Xinrun Xu, and 1 others. 2024. Towards general computer control: A multimodal agent for red dead redemption ii as a case study. In *ICLR 2024 Workshop on Large Language Model (LLM) Agents*.

Jianqiang Wan, Sibo Song, Wenwen Yu, Yuliang Liu, Wenqing Cheng, Fei Huang, Xiang Bai, Cong Yao, and Zhibo Yang. 2024. Omniparser: A unified framework for text spotting key information extraction and table recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15641–15653.

Bryan Wang, Yuliang Li, Zhaoyang Lv, Haijun Xia, Yan Xu, and Raj Sodhi. 2024a. Lave: Llm-powered agent assistance and language augmentation for video editing. In *Proceedings of the 29th International Conference on Intelligent User Interfaces*, pages 699–714.

Junyang Wang, Haiyang Xu, Haitao Jia, Xi Zhang, Ming Yan, Weizhou Shen, Ji Zhang, Fei Huang, and Jitao Sang. 2024b. Mobile-agent-v2: Mobile device operation assistant with effective navigation via multi-agent collaboration. *arXiv preprint arXiv:2406.01014*.

Junyang Wang, Haiyang Xu, Jiabo Ye, Ming Yan, Weizhou Shen, Ji Zhang, Fei Huang, and Jitao Sang. 2024c. Mobile-agent: Autonomous multi-modal mobile device agent with visual perception. *arXiv preprint arXiv:2401.16158*.

Shuai Wang, Weiwen Liu, Jingxuan Chen, Weinan Gan, Xingshan Zeng, Shuai Yu, Xinlong Hao, Kun Shao, Yasheng Wang, and Ruiming Tang. 2024d. Gui agents with foundation models: A comprehensive survey. *arXiv preprint arXiv:2411.04890*.

Weihan Wang, Qingsong Lv, Wenmeng Yu, Wenyi Hong, Ji Qi, Yan Wang, Junhui Ji, Zhuoyi Yang, Lei Zhao, Xixuan Song, and 1 others. 2023. Cogvlm: Visual expert for pretrained language models. *arXiv preprint arXiv:2311.03079*.

Xiaohan Wang, Yuhui Zhang, Orr Zohar, and Serena Yeung-Levy. 2024e. Videoagent: Long-form video understanding with large language model as agent. In *European Conference on Computer Vision*, pages 58–76. Springer.

Zhenhailong Wang, Haiyang Xu, Junyang Wang, Xi Zhang, Ming Yan, Ji Zhang, Fei Huang, and Heng Ji. 2025. Mobile-agent-e: Self-evolving mobile assistant for complex tasks. *arXiv preprint arXiv:2501.11733*.

Zhiyu Wu, Xiaokang Chen, Zizheng Pan, Xingchao Liu, Wen Liu, Damai Dai, Huazuo Gao, Yiyang Ma, Chengyue Wu, Bingxuan Wang, and 1 others. 2024. Deepseek-vl2: Mixture-of-experts vision-language models for advanced multimodal understanding. *arXiv preprint arXiv:2412.10302*.

Tianbao Xie, Danyang Zhang, Jixuan Chen, Xiaochuan Li, Siheng Zhao, Ruisheng Cao, Toh Jing Hua, Zhoujun Cheng, Dongchan Shin, Fangyu Lei, Yitao Liu, Yiheng Xu, Shuyan Zhou, Silvio Savarese, Caiming Xiong, Victor Zhong, and Tao Yu. 2024. Osworld: Benchmarking multimodal agents for open-ended tasks in real computer environments. *Preprint*, arXiv:2404.07972.

Mingzhe Xing, Rongkai Zhang, Hui Xue, Qi Chen, Fan Yang, and Zhen Xiao. 2024. Understanding the weakness of large language model agents within a complex android environment. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 6061–6072.

Yifan Xu, Xiao Liu, Xueqiao Sun, Siyi Cheng, Hao Yu, Hanyu Lai, Shudan Zhang, Dan Zhang, Jie Tang, and Yuxiao Dong. 2024. Androidlab: Training and systematic benchmarking of android autonomous agents. *arXiv preprint arXiv:2410.24024*.

Zhao Yang, Jiaxuan Liu, Yucheng Han, Xin Chen, Zebiao Huang, Bin Fu, and Gang Yu. 2023. Appagent: Multimodal agents as smartphone users. *arXiv preprint arXiv:2312.13771*.

Jiabo Ye, Haiyang Xu, Haowei Liu, Anwen Hu, Ming Yan, Qi Qian, Ji Zhang, Fei Huang, and Jingren Zhou. 2024. mplug-owl3: Towards long image-sequence understanding in multi-modal large language models. *arXiv preprint arXiv:2408.04840*.

Qinghao Ye, Haiyang Xu, Guohai Xu, Jiabo Ye, Ming Yan, Yiyang Zhou, Junyang Wang, Anwen Hu, Pengcheng Shi, Yaya Shi, and 1 others. 2023a. mplug-owl: Modularization empowers large language models with multimodality. *arXiv preprint arXiv:2304.14178*.

10

Qinghao Ye, Haiyang Xu, Jiabo Ye, Ming Yan, Haowei Liu, Qi Qian, Ji Zhang, Fei Huang, and Jingren Zhou. 2023b. mplug-owl2: Revolutionizing multi-modal large language model with modality collaboration. *arXiv preprint arXiv:2311.04257*.

Ori Yoran, Samuel Joseph Amouyal, Chaitanya Malaviya, Ben Bogin, Ofir Press, and Jonathan Berant. 2024. Assistantbench: Can web agents solve realistic and time-consuming tasks? *Preprint*, arXiv:2407.15711.

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

Chaoyun Zhang, Liqun Li, Shilin He, Xu Zhang, Bo Qiao, Si Qin, Minghua Ma, Yu Kang, Qingwei Lin, Saravan Rajmohan, Dongmei Zhang, and Qi Zhang. 2024a. UFO: A UI-Focused Agent for Windows OS Interaction. *arXiv preprint arXiv:2402.07939*.

Jiwen Zhang, Jihao Wu, Yihua Teng, Minghui Liao, Nuo Xu, Xiao Xiao, Zhongyu Wei, and Duyu Tang. 2024b. Android in the zoo: Chain-of-action-thought for gui agents. *arXiv preprint arXiv:2403.02713*.

Lu Zhang, Tiancheng Zhao, Heting Ying, Yibo Ma, and Kyusong Lee. 2024c. Omagent: A multi-modal agent framework for complex video understanding with task divide-and-conquer. *arXiv preprint arXiv:2406.16620*.

Boyuan Zheng, Boyu Gou, Jihyung Kil, Huan Sun, and Yu Su. 2024. Gpt-4v(ision) is a generalist web agent, if grounded. In *Forty-first International Conference on Machine Learning*.

Shuyan Zhou, Frank F Xu, Hao Zhu, Xuhui Zhou, Robert Lo, Abishek Sridhar, Xianyi Cheng, Yonatan Bisk, Daniel Fried, Uri Alon, and 1 others. 2023. Webarena: A realistic web environment for building autonomous agents. *arXiv preprint arXiv:2307.13854*.

Deyao Zhu, Jun Chen, Xiaoqian Shen, Xiang Li, and Mohamed Elhoseiny. 2023. Minigpt-4: Enhancing vision-language understanding with advanced large language models. *arXiv preprint arXiv:2304.10592*.

A Appendix

A.1 Experimental Details

This section provides additional details regarding the experimental setup and implementation choices used in Mobile-Agent-V.

A.1.1 Sliding Window Size Selection

In our experiments, the sliding window size was set to 4. While increasing the window size to 5 is also feasible, experimental analysis demonstrated that the performance improvement was marginal, while the computational cost increased due to the higher token consumption. Therefore, we adopted a window size of 4 as a balanced trade-off between efficiency and performance.

A.1.2 Video Similarity Computation

To compute the similarity between video frames, we employed a simple yet effective approach based on pixel-wise differences. Given two frames I_1 and I_2, we first converted them to grayscale representations:

$$I_1' = \text{grayscale}(I_1), \quad I_2' = \text{grayscale}(I_2) \quad (9)$$

Next, we computed the absolute difference between the two grayscale images:

$$D = \text{absdiff}(I_1', I_2') \quad (10)$$

Finally, the similarity score S was obtained by counting the number of nonzero pixels in D:

$$S = \frac{\text{np.count\_nonzero}(D)}{\text{total pixels}} \quad (11)$$

This method effectively captures differences between frames while maintaining computational efficiency.

A.1.3 Frame Similarity Threshold Selection

As described in the main text, the similarity threshold f_s was adjusted according to the characteristics of different applications. For instance, in the *Settings* app, where UI changes are primarily text-based, we set $f_s = 0.3$ to ensure that more informative frames were retained. Conversely, for the *Weather* app, where UI elements exhibit significant visual variations, a higher threshold of $f_s = 0.5$ was used to prevent excessive redundant frame extraction.

A.1.4 Step Limitations and Task Termination Criteria

To ensure fair evaluation and prevent infinite loops, we imposed an upper bound on the number of execution steps:

- Basic tasks: 10-step limit.

- Standard tasks: 15-step limit.

Action	Parameter	Description
Click	id	The "id" represents the numeric identifier of the detection box to be clicked.
Click_text	text	The "text" specifies the target text to be clicked, used only when no detection box or corresponding ID exists at the target location.
Scroll	direction	The "direction" can be either "up" or "down," allowing the agent to scroll the screen accordingly.
Type	text	The "text" parameter defines the content to be entered into a text field.
Back	None	Returns to the previous screen.
Home	None	Navigates to the home screen.
Done	None	Signals task completion.

Table 4: Action space definition for Mobile-Agent-V.

- Complex tasks: 20-step limit.

If an agent reached the step limit without successfully completing the task, the attempt was deemed a failure. Additionally, if a framework executed the required action but continued performing unnecessary operations beyond the instruction's scope, it was also considered a failure.

A.1.5 Video Frame Concatenation for Visualization

To simplify interpretation, video frames were concatenated in a row-wise manner. Each frame within the sliding window was indexed to aid the video agent in tracking its progress. In instances where fewer than four frames were available, only the existing frames (up to three) were concatenated. The final frame in each sequence was distinctly marked as the termination state, guiding the decision agent to stop at the correct point.

A.1.6 Action Space Definition

Mobile-Agent-V utilizes the same action space as Mobile-Agent-V2. Unlike Mobile-Agent-V2, which employs OCR and segmentation models to identify interaction coordinates, Mobile-Agent-V uses the Set of Mark (SoM) approach to decrease context length. To address potential XML parsing issues in certain UI pages, a supplementary click-by-text operation was introduced. A complete outline of the action space is provided in Table 4.

A.2 Prompt

Tables 5, 6, and 7 display the prompts used by the deep-reflection agent, decision agent, and video agent, respectively.

A.3 Benchmark Details

A.3.1 Evaluation Tasks of Mobile-Knowledge

Table 9 presents a comprehensive breakdown of benchmark tasks, categorized by application. This structure evaluates Mobile-Agent-V's proficiency in interpreting, aligning, and executing user instructions of varying complexity. The benchmark differentiates between video-aligned and video-misaligned instructions, testing the framework's robustness against linguistic variations and its adaptability to real-world user interactions.

A.3.2 Evaluation Tasks of AndroidWorld-Knowledge

Table 8 shows the task names from Android World in AndroidWorld-Knowledge.

A.3.3 Metrics

The following metrics characterize the evaluation process:

- **Success Rate**: This metric represents the percentage of instructions that are fully completed, offering a comprehensive measure of the agent's capability in executing tasks from start to finish without errors. A high success rate indicates proficient end-to-end execution, underscoring the agent's overall effectiveness and reliability in automating tasks accurately and efficiently.

- **Completion Rate**: Completion Rate quantifies the proportion of individual steps executed within a given instruction, providing a more granular view of task progression.

System
You are an expert in mobile phone operation. I will upload two images below. The first image is a keyframe mosaic from an operation video, in which the completed task is "$\{I_v\}$"; the second image is a screenshot of the current status of the mobile phone.

On the mobile phone shown in the second image, the task to be completed is: "$\{I_u\}$". The user will perform the following operation:
{Operation from decision agent}

Now please observe whether this operation conforms to the operation path shown in the first image. If it conforms, please output "True", otherwise please modify the operation content according to the above json format.

The operation should be:
- Click (id): The "id" is the numeric serial number of the detection box you need to click.
- Click_text (text): The "text" is the text you need to click. This is only used when the detection box and the corresponding id do not exist at the location to be clicked.
- Scroll (direction): The "direction" selects from "up" and "down". You can scroll the page a certain distance in the specified direction.
- Type (text): The "text" is the content you need to enter.
- Back: You can use this operation to return to the previous page.
- Home: You can use this operation to return to the home page.
- Done: You can use this operation when the task is completed.
Note: If the operation history and current device can infer that the task has been completed, use Done.

You need to think in the following way:
1. Observe the operation of each step in the video (especially frame-3 and frame-4).
2. Anchor the position of the current device in the video.
3. Complete the current step according to the operation in the video.
Please output your thought about this step by step before you output your response. |
| **User** |
| <image: V_w><image: D_i> |

Table 5: The prompt for deep-reflection agent.

This metric is essential for understanding areas where the agent may excel or face challenges, particularly in the execution of sequential tasks. By analyzing completion rates, researchers and developers can identify specific steps that require optimization or redesign to enhance overall task completion.

- **Decision Accuracy**: This metric evaluates the precision of the agent's decision-making processes by comparing the number of correctly made decisions against the total number of decisions attempted. High decision accuracy reflects the agent's adeptness in selecting appropriate actions based on provided data, highlighting its ability to navigate complex decision spaces effectively.

- **Step Count**: Step Count provides insight into the number of actions the agent takes to ac-

complish a given instruction and acts as a measure of execution efficiency. By tracking the steps required for task completion, this metric aids in pinpointing inefficiencies and excessive actions that may hinder performance.

A.3.4 Screen Recording

All videos were captured using the built-in screen recording tool on a OnePlus 7 Pro test device. While the tool supports a maximum frame rate of 60 Hz, practical frame rates ranged between 30 Hz and 60 Hz, contingent upon the degree of UI changes. Interactions were manually performed at an average frequency of one action every 1–2 seconds. The videos were left unprocessed, free from edits such as acceleration or overlays, thus preserving their original state. Each benchmark instruction corresponds to a unique operation video, demonstrating the optimal path for task execution.

System

You are a mobile phone operation assistant. Below is a description of this conversation.

In the following part, I will upload a large image made up of many screenshots. These screenshots in this image are all from a screen recording of a mobile phone operation. I will tell you the task completed in the screen recording. You need to observe this screen recording.

Then, you need to complete a new task, which is related to the task in the screen recording. You need to combine the operation experience provided by the screen recording and gradually complete this task. I will upload the current screenshot of the device. There will be many detection boxes on this screenshot, and there will be a number in the upper left and lower right corners of the detection box. You need to perform operations on the current page. In order to better operate the phone, the following are the operation tools you can use:
- Click (id): The "id" is the numeric serial number of the detection box you need to click.
- Click_text (text): The "text" is the text you need to click. This is only used when the detection box and the corresponding id do not exist at the location to be clicked.
- Scroll (direction): The "direction" selects from "up", "down", "left", and "right". You can scroll the page a certain distance in the specified direction.
- Type (text): The "text" is the content you need to enter.
- Back: You can use this operation to return to the previous page.
- Home: You can use this operation to return to the home page.
- Done: You can use this operation when the task is completed.

You need to strictly follow the following json output format:
"Thought": You need to think about how to perform this operation on the current device based on the operation path in the video, "Operation": Select one from the operation tools, "Summary": Briefly summarize this operation

User during the first operation

The first image is the screen recording, in which the tasks are completed: $\{I_v\}$

The second image is the screenshot of the current device, in which you need to complete the following tasks: $\{I_u\}$

Note: You need to refer to the operation path in the video more than relying on your own operation experience. Because you may make mistakes.

Note: You need to refer to the operation path in the video more than relying on your own operation experience. Because you may make mistakes."
<image: V_w><image: D_i>

User during subsequent operations

The first image is the screen recording, in which the tasks are completed: $\{I_v\}$

The second image is the screenshot of the current device, in which you need to complete the following tasks: $\{I_u\}$

Here is your operation history:
Step-1: {operation 1}
Step-2: {operation 2}
......
Step-n: {operation n}

Note: If the operation history and current device can infer that the task has been completed, use Done.

Note: You need to refer to the operation path in the video more than relying on your own operation experience. Because you may make mistakes."
<image: V_w><image: D_i>

Table 6: The prompt for decision agent.

System
You are a mobile phone operation assistant. I will provide you with two images. The first image is a long picture of key frames from a mobile phone operation video, which shows a correct operation trajectory to complete the task: $\{I_v\}$. The second image is two screenshots before and after an operation from the user. The user want to complete the task: $\{I_u\}$. Please note that these two images are not necessarily the complete operation trajectories, they may only be part of the continuous operation.

Although the task shown in the video may not be exactly the same as the task the user needs to complete, there is a strong correlation between the two. So the user is referring to the operation in the video to complete this task.

Now you need to determine which frame of the video the user is in after the device is operated. You need to use a number to represent it. If the device is in the state between two frames, the previous frame is output.
If the device is not in any frame of the video, please output the number 0 to indicate an operation error and generate an error cause analysis.

You need to output in the following json format:
{"Thought": Your thought of current question, "Frame": a number, "Analysis": If Frame is 0, generate an error cause analysis, otherwise output null, "Need_Back": If Frame is 0, you need to think about how to get back on track. If you need to return to the previous page, please output true. If you need to continue to perform an operation on the current page to get back on track, please output false. If Frame is not 0, please output False directly.}

User
Here are the video and operation:
<image: V_w><image: D_i>

Table 7: The prompt for video agent.

Applications	Task Name
Expense	ExpenseAddMultiple, ExpenseAddMultipleFromGallery, ExpenseAddMultipleFromMarkor, ExpenseAddSingle, ExpenseDeleteDuplicates, ExpenseDeleteDuplicates2, ExpenseDeleteMultiple, ExpenseDeleteMultiple2, ExpenseDeleteSingle
Markor	MarkorAddNoteHeader, MarkorChangeNoteContent, MarkorCreateFolder, MarkorCreateNote, MarkorCreateNoteAndSms, MarkorCreateNoteFromClipboard, MarkorDeleteAllNotes, MarkorDeleteNewestNote, MarkorDeleteNote, MarkorEditNote, MarkorMergeNotes, MarkorMoveNote, MarkorTranscribeReceipt, MarkorTranscribeVideo
Recipe	RecipeAddMultipleRecipes, RecipeAddMultipleRecipesFromImage, RecipeAddMultipleRecipesFromMarkor, RecipeAddMultipleRecipesFromMarkor2, RecipeAddSingleRecipe, RecipeDeleteDuplicateRecipes, RecipeDeleteDuplicateRecipes2, RecipeDeleteDuplicateRecipes3, RecipeDeleteMultipleRecipes, RecipeDeleteMultipleRecipesWithConstraint, RecipeDeleteMultipleRecipesWithNoise, RecipeDeleteSingleRecipe, RecipeDeleteSingleWithRecipeWithNoise
SportsTracker	SportsTrackerActivitiesCountForWeek, SportsTrackerActivitiesOnDate, SportsTrackerActivityDuration, SportsTrackerLongestDistanceActivity, SportsTrackerTotalDistanceForCategoryOverInterval, SportsTrackerTotalDurationForCategoryThisWeek
Tasks	TasksCompletedTasksForDate, TasksDueNextWeek, TasksDueOnDate, TasksHighPriorityTasks, TasksHighPriorityTasksDueOnDate, TasksIncompleteTasksOnDate

Table 8: Tasks in AndroidWorld-Knowledge.

APP	Level	Video Instruction & Video-Aligned User Instruction	Video-Misaligned User Instruction
Phone	Basic	Help me dial 123.	Help me dial 321.
	Normal	Please turn on the call recording for me.	Please view all call recording for me.
	Advanced	Help me add the mobile number 1234567890 to the blacklist.	Help me add the mobile number 9876543210 to the whitelist.
Messages	Basic	Help me set up messages and notifications to be displayed together in Messages.	Help me set up messages and notifications not to be displayed together in Messages.
	Normal	Please send a message to 123456 with text "Hello"	Please send a message to 9876543210 with text "Goodbye".
	Advanced	Send a message to 123456 with my current location information.	Send a message to 987654 with my contact card.
Setting	Basic	Help me turn off the auto brightness in Setting.	Help me turn on the auto brightness in Setting.
	Normal	Help me turn off the status bar network speed display.	Help me turn off the status bar NFC display.
	Advanced	Help me open three-finger screenshots.	Help me open three-finger touch and hold.
Photo	Basic	Help me turn on the shared albums setting in Photos.	Help me turn off the shared albums setting in Photos.
	Normal	Help me clear recently deleted photos.	Help me restore recently deleted photos.
	Advanced	Help me set up not to record location when taking photos.	Help me set up not to record properties when taking photos.
Manager	Basic	Help me turn on the App cleaner reminder in Phone Manager.	Help me turn off the App cleaner reminder in Phone Manager.
	Normal	Help me turn on the automatic phone call for help.	Help me turn on the automatic phone call for help and countdown sound.
	Advanced	Help me clean up QQ's storage.	Help me clean up WhatsApp's storage.
Recorder	Basic	Help me start recording.	Help me stop recording.
	Normal	Help me change the audio format of my recording.	Help me turn on the cloud recording.
	Advanced	Help me show recently deleted recordings.	Help me show call recordings.
Files	Basic	Help me view photos in My Files.	Help me view videos in My Files.
	Normal	Help me create a new tag named "test".	Help me create a new tag named "mobile".
	Advanced	Help me turn on the option to show hidden files.	Help me turn off the option to show hidden files.
Clock	Basic	Help me start stopwatch in Clock.	Help me reset stopwatch in Clock.
	Normal	Help me set the gesture to turn off the alarm to swipe up.	Help me set the gesture to turn off the alarm to press button.
	Advanced	Help me delete the last city of the current world clock and add London.	Help me delete the first city of the current world clock and add New York.
Weather	Basic	Help me turn on the meteorological alert setting in Weather.	Help me turn off the meteorological alert setting in Weather.
	Normal	Help me turn on the rain reminder.	Help me turn off the rain reminder.
	Advanced	Help me turn on the UV intensity display and view the UV intensity at your current location.	Help me turn on the Sunset display and view the sunset at your current location.
Calendar	Basic	Help me turn on fixed time zone setting in Calendar.	Help me turn off fixed time zone setting in Calendar.
	Normal	Help me turn on calendar meeting reminders.	Help me turn on fixed time zone.
	Advanced	Help me subscribe to horoscope and choose Aries.	Help me subscribe to today in history.

Table 9: Tasks in Mobile-knowledge.