ARIA-UI: VISUAL GROUNDING FOR GUI INSTRUCTIONS

Anonymous authors
Paper under double-blind review

ABSTRACT

Digital agents for automating tasks across different platforms by directly manipulating the GUIs are increasingly important. For these agents, grounding from language instructions to target elements remains a significant challenge due to reliance on HTML or AXTree inputs. In this paper, we introduce Aria-UI, a large multimodal model specifically designed for GUI grounding. Aria-UI adopts a pure-vision approach, eschewing reliance on auxiliary inputs. To adapt to heterogeneous planning instructions, we propose a scalable data pipeline that synthesizes diverse and high-quality instruction samples for grounding. To handle dynamic contexts in task performing, Aria-UI incorporates textual and text-image interleaved action histories, enabling robust context-aware reasoning for grounding. Aria-UI sets new state-of-the-art results across offline and online agent benchmarks, outperforming both vision-only and AXTree-reliant baselines. We release all training data and model checkpoints to foster further research.

1 INTRODUCTION

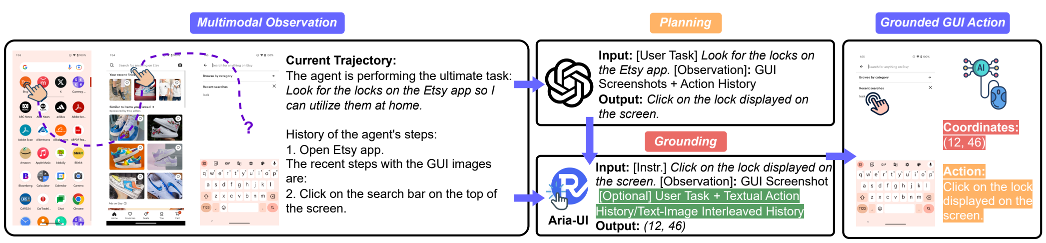

Figure 1: The two-stage task performing process for general GUI agents. Aria-UI serves as a robust grounding model to make the planned actions truly happen.

Collection	#Web Img.	#Mobile Img.	#Desktop Img.	Input Text	Supervision	Open Source	Action History	#Elements	#Samples
Ferret-UI-AMP	/	84K	/	Human Ann.	Point Coordinates	✗	✗	-	160K
CogAgent-CCS400K	400K	/	/	HTML Text	Point Coordinates	✗	✗	70M	-
UGround-Web-Hybrid	773K	/	/	HTML Attr. + Refer. Caption	Point Coordinates	✗	✗	18.1M	9M
UGround-Web-Direct	408K	/	/	Refer. Caption	Point Coordinates	✗	✗	408K	408K
SeeClick	270K	/	/	HTML Text	Point Coordinates	✓	✗	3.3M	3.3M
GUIEnv-local	73K	9K	/	HTML Text	Point Coordinates	✓	✗	700K	700K
Aria-UI Collection	**173K**	**104K**	**1.3K**	**Diversified Instr.**	**Refer. Caption + Point Coordinates**	✓	✓	**3.9M**	**11.5M**

Table 1: Grounding data of Aria-UI compared to existing collections.

The rapid expansion of graphical user interfaces (GUIs) across web, desktop and mobile platforms has made them indispensable for digital interactions. From completing daily tasks like shopping or booking tickets to complex professional workflows, GUI agents play a critical role in automating these processes. As illustrated in Figure 1, a typical GUI agent operates in two stages: planning and grounding. In the planning stage, the agent generates action decisions to accomplish the user's task based on the current screen state as its observation. In the grounding stage, the agent is tasked with locating and interacting with the target element as referred in the instructions provided by planning, thus make actions truly happen in the environment.

While efforts have been put to improve the planning of large multimodal models (LMMs) with CoT Yao et al. (2022b); Wei et al. (2022), and inference-time scaling Saha et al. (2024), effectively grounding GUI elements from language remains a significant challenge. The problem is compounded by the diverse visual layouts across diverse devices, wide variability in planned instructions, and the dynamic nature of task execution in real-world environments, all of which demand robust, adaptable, and efficient solutions.

The basic grounding method involves leveraging HTML or accessibility trees (AXTress, or A11y) to identify the target element. However, feeding long textual contexts of the tree often leads to inefficiencies, hallucination, and biases due to missing information in the tree. The absence of visual input further limits the method's ability to address instructions requiring visual or positional cues. Set-of-Mark Yang et al. (2023) combines visual and tree tag information. However, its reliance on HTML or AXTrees limits flexibility in diverse environments, as platform standards are inconsistent and, particularly on mobile and desktop, the quality of AXTrees depend largely on app developers' implementation. Additionally, LMMs struggle to accurately select from numerous tags in images, constraining grounding performance Xie et al. (2024). To this end, building a pure-vision solution for GUI agent grounding is crucial.

Training an LMM for GUI instruction grounding is non-trivial. Existing LMMs are: 1) heavily skewed towards natural images due to data biases. 2) rarely trained for grounding. While some models are trained with datasets like RefCOCO Kazemzadeh et al. (2014), these datasets are not aligned with GUI scenarios and are sparsely populated. Recently, some studies Cheng et al. (2024); Gou et al. (2024) have leveraged LMMs' powerful vision and language capabilities, using public mobile- or web-sourced data as (GUI image, instruction, coordinates) tuples to train LMMs as grounding models. Despite their effectiveness, we identify two key limitations in these approaches: **(1) They overly depend on rigid instruction sources and formats**, mainly HTML or AXTree-based textual elements. This lack of diversity hinders their robustness in adapting to the flexible and heterogeneous instructions generated by task planners. **(2) They overlook the dynamic contextual information during task performing**, such as the action history, which can provide valuable references for more accurate element grounding.

In this paper, we introduce Aria-UI, a robust LMM designed specifically for GUI grounding. Aria-UI is built upon Aria Li et al. (2024a), the state-of-the-art multimodal MoE model with 3.9B activated parameters. Aria-UI adopts a pure-vision approach, avoiding reliance on AXTree-like inputs while achieving superior grounding accuracy across diverse tasks and platforms.

By addressing the core limitations of existing methods, we propose two key contributions in Aria-UI. For the challenge of rigid instructions, we design a large-scale, diverse data synthesis pipeline from our Common Crawl collection and public available data. This pipeline first leverages strong LMMs to generate detailed and accurate element captions and then utilizes an LLM to create diverse, human-like instructions that align with potential interactions based on these captions. We further incorporate the high-quality captions as additional supervision during training, enabling the model to better associate diverse instructions with their corresponding elements. For the challenge of ignoring dynamic contexts, we further leverage textual or text-image interleaved action history from trajectory data for training. This equips Aria-UI with robust grounding capabilities, enabling it to perform effectively in dynamic, multi-step real-world task scenarios.

To summarize, our contributions are:

- We propose a novel approach to address the challenge of rigid instructions with a scalable, data-centric pipeline. It generates high-quality and diverse (element caption, instruction) samples from Common Crawl and publicly available data, enabling Aria-UI to generalize effectively across diverse instructions in different environments.

- Aria-UI introduces innovative designs for incorporating dynamic action history in textual or interleaved text-image formats. The improvements allow Aria-UI to ground elements more effectively in dynamic, multi-step task scenarios, especially under zero-shot settings.

- We conduct comprehensive evaluations on extensive benchmarks including both offline and online agent tasks, showcasing Aria-UI's state-of-the-art performance. Notably, Aria-UI achieves higher grounding accuracy and task success rates compared to both vision-only and AXTree-reliant baselines.

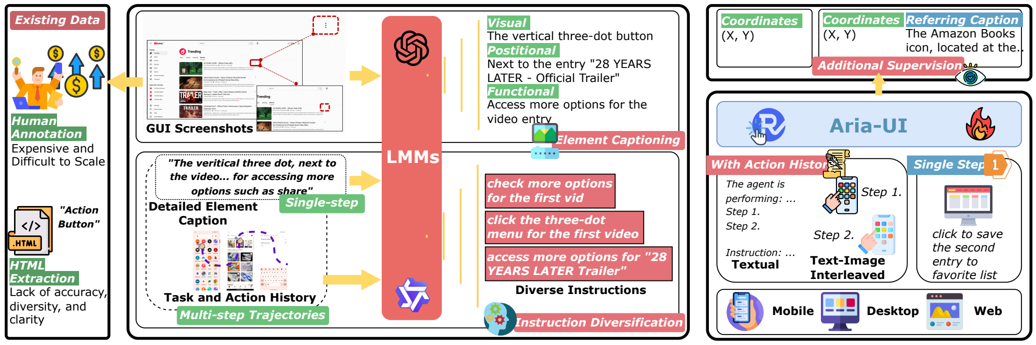

Figure 2: The overall data and training pipeline for Aria-UI.

2 Method

Aria-UI is designed to seamlessly integrate into the latest general-purpose multimodal GUI agent framework Zheng et al. (2024); Xie et al. (2024); Koh et al. (2024); Rawles et al. (2024a), serving as a robust grounding model. We outline a solution to the challenges from a scalable, data-centric approach, as shown in Figure 2. In Section 2.1.1, we detail the synthesizing of diverse grounding data. Section 2.1.2 discusses building grounding samples with task context for dynamic scenarios, and Section 2.2 explains Aria-UI 's training details.

2.1 Large-scale Diverse GUI Data Synthesizing

As summarized in Table 1, several existing methods have collected diverse corpus for GUI grounding. However, these corpora fail to effectively address GUI grounding for LMMs. They are either not open-source, too small, or lack coverage of all the major platforms. Moreover, they rely on rigid instruction sources and formats, from HTML extraction or specifically formatted referring caption. Additionally, they overlook the importance of the contextual information for grounding during dynamic task performing. We present how to solve these challenges by a data-centric approach with diverse data scaling from multiple platforms and context-aware data extension with trajectories.

2.1.1 Diverse Data Scaling from Multiple Platforms

We propose a two-stage pipeline to transform raw samples into high-quality and diverse element instructions for grounding training. At the first stage, we utilize a strong LMM (GPT-4o or Qwen2-VL-72B Wang et al. (2024a)) that takes element screenshots and text extracted from HTML as input for accurate and detailed element descriptions. To enhance accuracy and reduce hallucination, the model perceives two screenshots: (1) an isolated image of the element and (2) a zoomed-in view, where the element is highlighted with a red bounding box. Additionally, the HTML text and the screen position of the element are provided for reference. The model is then prompted to generate a detailed caption of the element, including its visual properties, functionality, positional relationships, and any other distinctive attributes. In the second stage, we utilize an LLM to generate natural language instructions that correspond to potential interactions with the elements, based on their detailed captions. For instance, for the caption *"The "subscribe" button, colored in bright red with white text and a bell icon, is positioned in the upper-right section of ChefMaria's cooking channel header, showing "2.3M" subscribers" underneath,"* the synthesized instruction could be *"subscribe to ChefMaria's channel."* To ensure diversity and expand the data volume, we produce three instructions for each element.

We apply our pipeline to three key GUI environments: web, desktop, and mobile, each with distinct challenges and characteristics.

Web. Web data, with its diversity and dynamic rendering, is ideal for expanding GUI grounding datasets with varied element samples in size, type, and resolution. We leverage the latest collection of Common Crawl for data collection. We build a rigorous data curation and filtering pipeline to produce high-quality samples. We first filter out harmful webpages using fastText Bojanowski et al. (2017). Subsequently, we identify and select interactive elements by checking the HTML attributes. Considering that LMMs have acquired fundamental OCR skills during pretraining, we prioritize graphical elements over text-based elements. To reflect real-world grounding tasks in complex, element-rich environments, we heuristically retain webpages containing more than 20 valid elements. We use Playwright to render these webpages at 1920×1080 and 2440×1600 resolutions to accommodate common resolution requirements. We gather a diverse set of 173K webpages containing 2M elements through the procedure. With the data pipeline, we build detailed caption and instructions for each element, and result in 6M high-quality and diverse instruction samples in total.

Desktop. Since desktop environment is less scalable and human annotation costs high, desktop data has remained scarce. OmniACT Kapoor et al. (2024) manually annotated 7.3K instruction-grounding pairs. However, creating an automated data scaling pipeline for desktop remains a challenge. To mitigate the research gap, we develop a traverse agent powered by an LMM to explore the OS environment for data collecting. We build the traverse agent on OSWorld Xie et al. (2024) with Gemini 1.5 Flash. Leveraging the accessibility tree, the agent selects the next element to click in each screen state, aiming to reach previously unexplored screens. We equip the agent with a simple memory mechanism and guide its exploration through a heuristic depth-first search. We collect all screenshots and the corresponding A11y to parse all elements. Using this automated pipeline, we collected 15K elements tailored for desktop environment. We then utilize the data pipeline to extend the samples to 45K by generating diverse instructions.

Mobile. Since automated GUI agents for mobile environments were explored earlier, a substantial amount of open-source data has been accumulated for mobile environment. Currently, the largest-scale grounding dataset for mobile is AMEX Chai et al. (2024), which provides 104K screenshots and 1.6M elements. While AMEX provides a large-scale dataset, it has only 712K elements with basic textual descriptions extracted from accessibility tags, and merely 3K elements are paired with human-like instructions. To address this gap, we regenerate high-quality caption and instruction samples with the data pipeline for AMEX, improving the training effectiveness while maintaining the same data volume.

Public Data. To further expand our grounding corpus and introduce more diverse sources for GUI images and instructions, we incorporate the following public datasets: 3M Web and 273K mobile elements from SeeClick training data Cheng et al. (2024); Li et al. (2020b;a), 15K mobile elements from Bai et al. (2021), 748k Web elements from GUICourse Chen et al. (2024), 131K desktop elements from OmniAct Kapoor et al. (2024), and 693K Web and mobile elements from AutoGUI [1].

2.1.2 Context-aware Data Extension from Trajectories

Accurately and efficiently performing grounding tasks within the dynamic context of real-world environments is a crucial capability for GUI agents. Despite its importance, existing approaches largely focus on grounding tasks under a single-step setting, where LMMs are trained to infer grounding results based only on the current state and instruction. Such approaches overlook the dynamic nature of GUI grounding and the critical role of context in real-world scenarios. For example, after executing a *TYPE* action, the next grounding step is likely associated with an *ENTER* or *SUBMIT* button. Similarly, in multi-step tasks that involve navigating through a multi-layered menu to locate a target entry, there is a strong contextual relationship between consecutive grounding actions. Leveraging such contextual information enriches the grounding context and aids the model in avoiding bias, thereby enhancing grounding performance.

We utilize publicly available agent trajectories to simulate grounding tasks with contexts. We focus on constructing two types of contextual setups: (1) textual action history and (2) text-image-interleaved history. The text-based setup incorporates the ultimate task along with prior action histories, and the text-image-interleaved setup extends this by including N historical screen state images, providing richer contextual cues and training the model to understand multimodal interaction history. Notably, most trajectory data only includes basic sequential information, such as the click coordinates, thus lacks comprehensive stepwise instruction semantics. To address this, we augment all grounding steps within the trajectory data using the proposed data pipeline to generate detailed stepwise instructions. For non-grounding actions, we encode instructions (e.g., SWIPE and TYPE) using rule-based methods for natural language formats. For the interleaved setting, we collect data as per $N = [1, 2, 3]$, and for the text-based setting, we input all historical actions in text. Finally we collect 992K samples with the

[1] https://huggingface.co/AutoGUI

trajectories from GUI-Odyssey Lu et al. (2024), Android in the Zoo Zhang et al. (2024d), Android Control Li et al. (2024b), Android in the Wild Rawles et al. (2024b) and AMEX Chai et al. (2024).

2.2 Model Architecture

Method	Mobile		Desktop		Web		Avg.
	Text	Icon/Widget	Text	Icon/Widget	Text	Icon/Widget	
GPT-4	22.6	24.5	20.2	11.8	9.2	8.8	16.7
GPT-4o	20.2	24.9	21.1	23.6	12.2	7.8	18.1
CogAgent	67.0	24.0	74.2	20.0	70.4	28.6	49.6
SeeClick	78.0	52.0	72.2	30.0	55.7	32.5	55.8
Qwen2-VL	75.5	60.7	76.3	54.3	35.2	25.7	55.3
UGround	82.8	60.3	82.5	63.6	80.4	70.4	74.1
Aria-UI	**92.3**	**73.8**	**93.3**	**64.3**	**86.5**	**76.2**	**82.4**

Table 2: Results on ScreenSpot. We report element accuracy and the micro average results.

We build Aria-UI with the state-of-the-art multimodal MoE model, Aria Li et al. (2024a). We leverage two strengths from Aria for GUI agents: 1) Aria is multimodal-native, built for better understanding of complex and interleaved contexts; 2) with only 3.9B activated parameters, Aria shows even faster inference speed than 7B dense models.

2.2.1 Ultra Resolution Support

With the shift from 1080p to 2K resolutions on computers and mobile devices, training grounding LMMs at high resolutions has become essential. Aria originally supports high-resolution images up to 980×980, which we extend to a maximum of 3920×2940 on Aria-UI by splitting the image into smaller blocks, significantly increasing the range of image sizes to handle. To maintain positional accuracy, we take inspiration from NaViT Dehghani et al. (2024) to place padding before resizing for keeping the original screenshot ratio.

2.3 Training and Inference Paradigm

We train Aria-UI following a two-phase procedure. We first leverage all the single-step grounding data to train the foundation GUI grounding capability of Aria-UI. Specifically, Aria-UI is tasked with generating grounding answers given the prompt *"Given a GUI image, what are the relative (0-1000) pixel point coordinates for the element corresponding to the following instruction or description: [...]"*. We follow Gou et al. (2024) to group all the samples for the same GUI image into a multi-turn conversation format. Then, context-aware data with both text-based and text-and-image-interleaved history settings are fed into the model to further enhance the grounding capability under the dynamic setting. For this phase, we add extra 20% samples from the single-step data to keep the generic grounding capability and avoid over-fitting.

During inference, Aria-UI outputs the grounded pixels coordinates normalized to $[0, 1000]$. Since Aria-UI is also trained with context-aware trajectories, it can take historical agent actions and grounding actions as chat history, formulating a stronger grounding system in dynamic environments.

3 Experiments

We testify the performances of Aria-UI via extensive experiments including single-step grounding, grounding under offline agent trajectories and grounding in dynamic online agent environments.

3.1 GUI Grounding Evaluation

We first examine Aria-UI's foundational GUI grounding capabilities on ScreenSpot Cheng et al. (2024). The benchmark compasses six subsets spanning over two types of elements and three major platforms. Each test entry provides a unique GUI image and a human-annotated instruction for locating a specific element. The typical resolution for mobile and web subsets is 2k, and for desktop samples it is 540p. We include the state-of-the-art UGround Gou et al. (2024), with previous

Models	AndroidControl-Low		AndroidControl-High		GUI-Odyssey	
	Grounding	Task SR	Grounding	Task SR	Grounding	Task SR
Zero-shot						
GPT-4o	16.36	5.12	10.36	2.84	19.66	0.05
Qwen2-VL	64.24	32.53	30.32	4.08	49.56	2.00
SeeClick	45.55	17.72	20.17	4.29	45.19	1.45
UGround	-	-	-	-	50.25	2.02
Aria-UI	**79.70**	**54.39**	**35.12**	**5.95**	**64.81**	**5.28**
W. Training Set						
UGround	74.28	46.85	37.98	9.15	-	-
Aria-UI	85.71	66.30	41.78	9.97	84.57	31.87
Aria-UI$_{TH}$	**87.69**	**67.33**	**43.16**	**10.17**	86.75	36.47
Aria-UI$_{IH}$	87.20	67.26	42.97	10.10	**87.02**	**37.30**

Table 3: Results for offline mobile agent evaluation. We report element accuracy for grounding and the task success rate. For AndroidControl-High, GPT-4o serves as the planner to generate stepwise instructions for all methods.

grounding models SeeClick Cheng et al. (2024) and CogAgent Hong et al. (2024) as baselines. We also include generic LMMs – GPT-4, GPT-4o and Qwen2-VL Wang et al. (2024a).

From the results in Table 2, Aria-UI achieves the highest average accuracy (82.4%) across all subsets, demonstrating its superior grounding performance. Aria-UI achieves a significant margin over the state-of-the-art UGround, particularly excelling in tasks for textual elements. The results showcase Aria-UI's robustness and generalizability across diverse platforms and element types.

3.2 Offline Agent Evaluation

Input	Planner	Grounding	Cross-Task	Cross-Website	Cross-Domain	Avg.
Image + HTML Tree	GPT-4	Choice	46.4	38.0	42.4	42.3
	GPT-4	SoM	29.6	20.1	27.0	25.6
Image	GPT-4	SeeClick	29.6	28.5	30.7	29.6
	GPT-4	UGround	45.1	44.7	44.6	44.8
	GPT-4	OmniParser	42.4	41.0	45.4	42.9
	GPT-4o	SeeClick	32.1	33.1	33.5	32.9
	GPT-4o	UGround	47.7	46.0	46.6	46.8
	GPT-4o	**Aria-UI**	**56.1**	**57.0**	**59.5**	**57.5**
	GPT-4o	**Aria-UI**$_{TH}$	**57.6**	**58.0**	**61.2**	**58.9**
	GPT-4o	**Aria-UI**$_{IH}$	**57.6**	**57.7**	**61.4**	**58.9**

Table 4: Results on Multimodal-Mind2Web, with grounding element accuracy reported. None of the methods adopted the training split, therefore we exhibit a fully zero-shot out-of-distribution evaluation.

Input	Planner	Grounding	AndroidWorld	MobileMiniWob++
AXTree	GPT-4-Turbo	Choice	30.6	59.7
	Gemini 1.5 Pro	Choice	19.4	57.4
Image + AXTree	GPT-4-Turbo	SoM	25.4	67.7
	Gemini 1.5 Pro	SoM	22.8	40.3
Image	GPT-4-Turbo	UGround	31.0	-
	GPT-4o	UGround	32.8	48.4
	GPT-4o	**Aria-UI**	**39.7**	**60.4**
	GPT-4o	**Aria-UI**$_{TH}$	**44.8**	-

Table 5: Task success rate results for online mobile and Web agents on AndroidWorld and MobileMiniWob++.

Mobile Agents. We further testify how Aria-UI performs under an offline dynamic setting, where the model is required to provide grounding coordinates in agent task trajectories. We employ AndroidControl-Low Li et al. (2024b), GUI-Odyssey Lu et al. (2024) and AndroidControl-High, the

Models	OS	Calc	Impress	Writer	VLC	Thunderbird	Chrome	VSC	GIMP	Multi	Avg.
GPT-4o + SoM	20.83	0.00	6.77	4.35	6.53	0.00	4.35	4.35	0.00	3.60	4.59
CogAgent + SoM	4.17	2.17	0.00	4.34	6.53	0.00	2.17	0.00	0.00	0.00	0.99
GPT-4o + A11y	41.67	4.26	6.81	8.70	9.50	6.67	15.22	30.43	0.00	7.46	11.21
CogAgent	4.17	2.17	0.00	4.35	6.53	0.00	2.17	0.00	0.00	0.10	1.11
GPT-4o	8.33	0.00	6.77	4.35	16.10	0.00	4.35	4.35	3.85	5.58	5.03
GPT-4o + **Aria-UI$_{TH}$**	**25.00**	**4.26**	**15.32**	**8.70**	**30.06**	**26.67**	**23.80**	**21.74**	**19.23**	**8.55**	**15.15**

Table 6: OSWorld results. The top part denotes methods with both accessibility tree (A11y) and screenshot input, while the bottom part is for pure-vision methods that rely only on screenshots.

first two has human-annotated or generated stepwise instruction, while the last one only provides the user task, and needs an additional planner for stepwise instructions. We follow Li et al. (2024b); Gou et al. (2024) to utilize GPT-4o as the planner. We report element accuracy and the task success rate in Table 3. Specifically, we evaluate Aria-UI and the baselines on both zero-shot and training split-included settings. As we evaluate Aria-UI with agent trajectories, we extend the model with two variants: Aria-UI$_{TH}$ and Aria-UI$_{IH}$, for textual action history input and text-image interleaved history input, separately. We choose $N = 1$ for Aria-UI$_{IH}$ to include additional one GUI image from history during inference.

The results demonstrate the superior performance of Aria-UI across different evaluation settings and metrics. Specifically, Aria-UI and its variants consistently outperform existing baselines, with Aria-UI$_{TH}$ achieving peak performance of grounding accuracy and task success rate on AndroidControl, and Aria-UI$_{IH}$ achieving the best performances on GUI-Odyssey. Empirically, we found that the incorporation of historical actions, whether in text-only (TH) or text-image interleaved (IH) format, provides crucial context for accurate element grounding and task completion. In particular, we observe that the textual action history (Aria-UI$_{TH}$) strikes an effective balance between efficiency and performance compared to both the base model and Aria-UI$_{IH}$.

In summary, the significant performance gap between Aria-UI and existing approaches like SeeClick and UGround underscores the effectiveness of our proposed model in understanding and executing mobile interface interactions.

Web Agents. We evaluate how Aria-UI and its variants perform on multimodal Web agent tasks with the Multimodal-Mind2Web Deng et al. (2024) benchmark. The original training split is not included by Aria-UI and the baselines during the training stage, thus we form a fully zero-shot out-of-distribution scenario. Three subsets, cross-task, cross-website and cross-domain are employed for a comprehensive evaluation.

Shown in Table 4, Aria-UI and its variants significantly outperform all baselines across the three subsets, achieving an average accuracy of 57.5% for the base model and 58.9% for Aria-UI$_{TH}$ and Aria-UI$_{IH}$. Notably, Aria-UI$_{IH}$ demonstrates the strongest performance in the cross-website and cross-domain subsets, showcasing its robust ability to leverage historical multimodal context. The improvements over previous models, including UGround and SeeClick, underscore Aria-UI's effectiveness in handling zero-shot grounding tasks on diverse and unseen web interfaces.

3.3 Online Agent Evaluation

Method	Mobile		Desktop		Web		Avg.
	Text	Icon/Widget	Text	Icon/Widget	Text	Icon/Widget	
Aria-UI	92.3	73.8	93.3	64.3	86.5	76.2	82.4
(-) Ultra Resolution	87.5	61.1	70.6	40.0	53.5	40.3	61.1
(+) Visual CoT Prompting	93.8	59.8	80.4	51.4	73.0	57.8	71.4
(-) Aria-UI Data	89.0	60.7	78.3	34.3	79.6	52.9	68.7
(-) Diversified Instruction	88.3	67.2	83.0	57.1	82.2	63.1	74.9
(-) Refer. as Supervision	92.7	69.0	81.4	54.3	85.2	70.0	77.5

Table 7: Ablation study results on ScreenSpot.

Mobile and Web. We use AndroidWorld Rawles et al. (2024a) for online mobile agent evaluation in an Android emulator environment. The evaluation is fully based on success of the task by checking the system state of the virtual device. We also include the MobileMiniWob++ task collection provided by AndroidWorld, which adpats the Web agent environment MiniWob++ Liu et al. (2018) to AndroidEnv Toyama et al. (2021), the same environment as AndroidWorld. We evalute Aria-UI with the strongest baseline, UGround under the same M3A agent framework, compared with SoM and Choice methods that require AXTree input. We report task success rate, the most important metric for real agents in Table 5. Our observations are:

- In AndroidWorld, our approach achieves the best performance to date, with a task success rate of 44.8%, achieved by Aria-UI$_{TH}$. This surpasses the previous state-of-the-art method, UGround, as well as non-pure vision methods such as SoM and Choice, which rely heavily on AXTree input. The results highlight Aria-UI's superior ability to handle diverse element instructions in real-world settings, demonstrating its robustness and adaptability for pure-vision GUI agents.
- On MobileMiniWob++, Aria-UI outperforms UGround, and choice-based methods. Due to the simplicity of MiniWob++ layouts, GPT-4-Turbo with SoM achieves the highest performance. However, Aria-UI still demonstrates the highest scores with pure-vision input.

OSWorld. We further evaluate Aria-UI on the most up-to-date and complex computer use simulator benchmark, OSWorld Xie et al. (2024). Following the pure-vision agent framework in OSWorld, we place Aria-UI as the grounding model to work collaboratively with GPT-4o on the 369 real tasks provided. We compared Aria-UI with previous SOTA methods and summarize the task success rate in Table 6. With GPT-4o as planner and Aria-UI$_{TH}$ as the grounding model, we achieve the highest average task success rate of 15.15%, outperforming previous methods across all computer-use scenarios in OSWorld. Notably, it excels in tasks like VLC (30.06%), Chrome (23.80%), and Impress (15.32%), highlighting Aria-UI's strong performance in diverse, complex GUI tasks.

3.4 Ablation Study

We further testify how Aria-UI performs with ablation settings of the proposed components through the following perspectives:

Model Components.

- (-) Ultra Resolution. We remove the ultra resolution support for Aria-UI.
- (+) Visual CoT Prompting. We use visual CoT prompting during test time for Aria-UI. For example: *"Think step-by-step with visual clues before giving the answer."*

Training Data Ablation.

- (-) Aria-UI Pipeline Data. We remove the data from our pipeline during training.
- (-) Diversified Instruction. We directly use refer. caption as input and coordinates as output for training, removing the diversified instructions.
- (-) Refer. as Supervision. We use only coordinates for supervision for our pipeline data.

We summarize the ablation results in Table 7. The results highlight the critical role of ultra resolution (Avg. 61.1) and Aria-UI data, particularly for Icon/Widget grounding. Removing diversified instruction or refer. as supervision degrades performance across platforms, due to weak alignment between instruction, refer. caption and grounding coordinates. We also found that adding CoT improves text-based tasks on mobile but struggles with others, caused by noise in visual reasoning.

4 Related Work

Vision-language Grounding with Large Multimodal Models. Foundational approaches for vision-language grounding, such as Zou et al. (2023); Liu et al. (2023); Li et al. (2023), integrate CLIP with specialized vision models to tackle language-guided grounding tasks. To address the limitations in complex reasoning scenarios, researchers have begun leveraging LMMs Liu et al. (2024); Dai et al. (2023); Shao et al. (2024) as a promising direction. Notable works Peng et al. (2023); Pi et al. (2023); Wang et al. (2024b) train LMMs to respond to fine-grained language instructions by grounding them in specific visual regions, while general-purpose models Bai et al. (2023); Li et al. (2024a) incorporate grounding as a core function during training. Additionally, significant advances in spatial information processing Zhang et al. (2023b); Chen et al. (2023); Zhang et al. (2023c); You et al. (2023); Zhang et al. (2024b) have enhanced regional visual comprehension capabilities. Despite these advancements, these methods, while effective for natural images, face challenges when applied to GUI screenshots due to insufficient specialized training data.

General GUI Agents. Automating GUI operations with capable agents has become a trending research area that leverages LMMs. Existing efforts have been put to desgin autonomous agents for complex task completion on mobile Rawles et al. (2024a); Bai et al. (2024); Li et al. (2024c); Zhang et al. (2023a); Wen et al. (2024); Nong et al. (2024); You et al. (2024); Li et al. (2024d), Web Koh et al. (2024); Yao et al. (2022a); Zhou et al. (2023); Lai et al. (2024); He et al. (2024); Abuelsaad et al. (2024); Ma et al. (2023); Zhang et al. (2024c) and desktop Xie et al. (2024); Wu et al. (2024); Gao et al. (2023); Zheng et al. (2023); Zhang et al. (2024a); Niu et al. (2024) environments. These methods initially relied on HTML or AXTrees for element grounding to perform actions. Recently, several notable studies Cheng et al. (2024); Gou et al. (2024) have proposed developing pure vision-based GUI grounding models with LMMs. However, due to their lack of instruction diversity and insufficient consideration of dynamic context, these approaches have delivered sub-optimal performances.

5 Conclusion

In this paper, we introduced Aria-UI, a robust LMM for GUI grounding across diverse environments. We designed a two-stage data pipeline for high-quality and diverse GUI grounding data from multiple platforms. We further incorporated dynamic action history as effective cues for stronger grounding capabilities in real-world environments. As a scalable and data-centric method, Aria-UI outperforms existing methods on all evaluated benchmarks, with both offline and online agent tasks. The model demonstrates strong zero-shot generalization across platforms, establishing Aria-UI as a powerful solution for universal GUI grounding.

References

Tamer Abuelsaad, Deepak Akkil, Prasenjit Dey, Ashish Jagmohan, Aditya Vempaty, and Ravi Kokku. Agent-e: From autonomous web navigation to foundational design principles in agentic systems. *arXiv preprint arXiv:2407.13032*, 2024.

Chongyang Bai, Xiaoxue Zang, Ying Xu, Srinivas Sunkara, Abhinav Rastogi, Jindong Chen, et al. Uibert: Learning generic multimodal representations for ui understanding. *arXiv preprint arXiv:2107.13731*, 2021.

Hao Bai, Yifei Zhou, Mert Cemri, Jiayi Pan, Alane Suhr, Sergey Levine, and Aviral Kumar. Digirl: Training in-the-wild device-control agents with autonomous reinforcement learning. *arXiv preprint arXiv:2406.11896*, 2024.

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

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. Enriching word vectors with subword information. *Transactions of the association for computational linguistics*, 5:135–146, 2017.

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 arXiv:2407.17490*, 2024.

Keqin Chen, Zhao Zhang, Weili Zeng, Richong Zhang, Feng Zhu, and Rui Zhao. Shikra: Unleashing multimodal llm's referential dialogue magic. *arXiv preprint arXiv:2306.15195*, 2023.

Wentong Chen, Junbo Cui, Jinyi Hu, Yujia Qin, Junjie Fang, Yue Zhao, Chongyi Wang, Jun Liu, Guirong Chen, Yupeng Huo, et al. Guicourse: From general vision language models to versatile gui agents. *arXiv preprint arXiv:2406.11317*, 2024.

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

Wenliang Dai, Junnan Li, Dongxu Li, Anthony Meng Huat Tiong, Junqi Zhao, Weisheng Wang, Boyang Li, Pascale Fung, and Steven Hoi. Instructblip: Towards general-purpose vision-language models with instruction tuning, 2023. URL `https://arxiv.org/abs/2305.06500`.

Mostafa Dehghani, Basil Mustafa, Josip Djolonga, Jonathan Heek, Matthias Minderer, Mathilde Caron, Andreas Steiner, Joan Puigcerver, Robert Geirhos, Ibrahim M Alabdulmohsin, et al. Patch n'pack: Navit, a vision transformer for any aspect ratio and resolution. *Advances in Neural Information Processing Systems*, 36, 2024.

Xiang Deng, Yu Gu, Boyuan Zheng, Shijie Chen, Sam Stevens, Boshi Wang, Huan Sun, and Yu Su. Mind2web: Towards a generalist agent for the web. *Advances in Neural Information Processing Systems*, 36, 2024.

Difei Gao, Lei Ji, Zechen Bai, Mingyu Ouyang, Peiran Li, Dongxing Mao, Qinchen Wu, Weichen Zhang, Peiyi Wang, Xiangwu Guo, et al. Assistgui: Task-oriented desktop graphical user interface automation. *arXiv preprint arXiv:2312.13108*, 2023.

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. *arXiv preprint arXiv:2410.05243*, 2024.

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

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

Raghav Kapoor, Yash Parag Butala, Melisa Russak, Jing Yu Koh, Kiran Kamble, Waseem Alshikh, and Ruslan Salakhutdinov. Omniact: A dataset and benchmark for enabling multimodal generalist autonomous agents for desktop and web. *arXiv e-prints*, pp. arXiv–2402, 2024.

Sahar Kazemzadeh, Vicente Ordonez, Mark Matten, and Tamara Berg. Referitgame: Referring to objects in photographs of natural scenes. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pp. 787–798, 2014.

Jing Yu Koh, Robert Lo, Lawrence Jang, Vikram Duvvur, Ming Chong Lim, Po-Yu Huang, Graham Neubig, Shuyan Zhou, Ruslan Salakhutdinov, and Daniel Fried. Visualwebarena: Evaluating multimodal agents on realistic visual web tasks. *arXiv preprint arXiv:2401.13649*, 2024.

Hanyu Lai, Xiao Liu, Iat Long Iong, Shuntian Yao, Yuxuan Chen, Pengbo Shen, Hao Yu, Hanchen Zhang, Xiaohan Zhang, Yuxiao Dong, et al. Autowebglm: Bootstrap and reinforce a large language model-based web navigating agent. *arXiv preprint arXiv:2404.03648*, 2024.

Dongxu Li, Yudong Liu, Haoning Wu, Yue Wang, Zhiqi Shen, Bowen Qu, Xinyao Niu, Guoyin Wang, Bei Chen, and Junnan Li. Aria: An open multimodal native mixture-of-experts model. *arXiv preprint arXiv:2410.05993*, 2024a.

Feng Li, Hao Zhang, Peize Sun, Xueyan Zou, Shilong Liu, Jianwei Yang, Chunyuan Li, Lei Zhang, and Jianfeng Gao. Semantic-sam: Segment and recognize anything at any granularity. *arXiv preprint arXiv:2307.04767*, 2023.

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

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

Yang Li, Jiacong He, Xin Zhou, Yuan Zhang, and Jason Baldridge. Mapping natural language instructions to mobile ui action sequences. *arXiv preprint arXiv:2005.03776*, 2020a.

Yang Li, Gang Li, Luheng He, Jingjie Zheng, Hong Li, and Zhiwei Guan. Widget captioning: Generating natural language description for mobile user interface elements. *arXiv preprint arXiv:2010.04295*, 2020b.

Zhangheng Li, Keen You, Haotian Zhang, Di Feng, Harsh Agrawal, Xiujun Li, Mohana Prasad Sathya Moorthy, Jeff Nichols, Yinfei Yang, and Zhe Gan. Ferret-ui 2: Mastering universal user interface understanding across platforms. *arXiv preprint arXiv:2410.18967*, 2024d.

Evan Zheran Liu, Kelvin Guu, Panupong Pasupat, Tianlin Shi, and Percy Liang. Reinforcement learning on web interfaces using workflow-guided exploration. *arXiv preprint arXiv:1802.08802*, 2018.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. *Advances in neural information processing systems*, 36, 2024.

Shilong Liu, Zhaoyang Zeng, Tianhe Ren, Feng Li, Hao Zhang, Jie Yang, Chunyuan Li, Jianwei Yang, Hang Su, Jun Zhu, et al. Grounding dino: Marrying dino with grounded pre-training for open-set object detection. *arXiv preprint arXiv:2303.05499*, 2023.

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 arXiv:2406.08451*, 2024.

Kaixin Ma, Hongming Zhang, Hongwei Wang, Xiaoman Pan, Wenhao Yu, and Dong Yu. Laser: Llm agent with state-space exploration for web navigation. *arXiv preprint arXiv:2309.08172*, 2023.

Runliang Niu, Jindong Li, Shiqi Wang, Yali Fu, Xiyu Hu, Xueyuan Leng, He Kong, Yi Chang, and Qi Wang. Screenagent: A vision language model-driven computer control agent. *arXiv preprint arXiv:2402.07945*, 2024.

Songqin Nong, Jiali Zhu, Rui Wu, Jiongchao Jin, Shuo Shan, Xiutian Huang, and Wenhao Xu. Mobileflow: A multimodal llm for mobile gui agent. *arXiv preprint arXiv:2407.04346*, 2024.

Zhiliang Peng, Wenhui Wang, Li Dong, Yaru Hao, Shaohan Huang, Shuming Ma, and Furu Wei. Kosmos-2: Grounding multimodal large language models to the world. *arXiv preprint arXiv:2306.14824*, 2023.

Renjie Pi, Jiahui Gao, Shizhe Diao, Rui Pan, Hanze Dong, Jipeng Zhang, Lewei Yao, Jianhua Han, Hang Xu, Lingpeng Kong, et al. Detgpt: Detect what you need via reasoning. *arXiv preprint arXiv:2305.14167*, 2023.

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

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, 2024b.

Swarnadeep Saha, Archiki Prasad, Justin Chih-Yao Chen, Peter Hase, Elias Stengel-Eskin, and Mohit Bansal. System-1. x: Learning to balance fast and slow planning with language models. *arXiv preprint arXiv:2407.14414*, 2024.

Hao Shao, Shengju Qian, Han Xiao, Guanglu Song, Zhuofan Zong, Letian Wang, Yu Liu, and Hongsheng Li. Visual cot: Unleashing chain-of-thought reasoning in multi-modal language models. *arXiv preprint arXiv:2403.16999*, 2024.

Daniel Toyama, Philippe Hamel, Anita Gergely, Gheorghe Comanici, Amelia Glaese, Zafarali Ahmed, Tyler Jackson, Shibl Mourad, and Doina Precup. Androidenv: A reinforcement learning platform for android. *arXiv preprint arXiv:2105.13231*, 2021.

Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, et al. Qwen2-vl: Enhancing vision-language model's perception of the world at any resolution. *arXiv preprint arXiv:2409.12191*, 2024a.

Wenhai Wang, Zhe Chen, Xiaokang Chen, Jiannan Wu, Xizhou Zhu, Gang Zeng, Ping Luo, Tong Lu, Jie Zhou, Yu Qiao, et al. Visionllm: Large language model is also an open-ended decoder for vision-centric tasks. *Advances in Neural Information Processing Systems*, 36, 2024b.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837, 2022.

Hao Wen, Yuanchun Li, Guohong Liu, Shanhui Zhao, Tao Yu, Toby Jia-Jun Li, Shiqi Jiang, Yunhao Liu, Yaqin Zhang, and Yunxin Liu. Autodroid: Llm-powered task automation in android. In *Proceedings of the 30th Annual International Conference on Mobile Computing and Networking*, pp. 543–557, 2024.

Zhiyong Wu, Chengcheng Han, Zichen Ding, Zhenmin Weng, Zhoumianze Liu, Shunyu Yao, Tao Yu, and Lingpeng Kong. Os-copilot: Towards generalist computer agents with self-improvement. *arXiv preprint arXiv:2402.07456*, 2024.

Tianbao Xie, Danyang Zhang, Jixuan Chen, Xiaochuan Li, Siheng Zhao, Ruisheng Cao, Toh Jing Hua, Zhoujun Cheng, Dongchan Shin, Fangyu Lei, et al. Osworld: Benchmarking multimodal agents for open-ended tasks in real computer environments. *arXiv preprint arXiv:2404.07972*, 2024.

Jianwei Yang, Hao Zhang, Feng Li, Xueyan Zou, Chunyuan Li, and Jianfeng Gao. Set-of-mark prompting unleashes extraordinary visual grounding in gpt-4v. *arXiv preprint arXiv:2310.11441*, 2023.

Shunyu Yao, Howard Chen, John Yang, and Karthik Narasimhan. Webshop: Towards scalable real-world web interaction with grounded language agents. *Advances in Neural Information Processing Systems*, 35:20744–20757, 2022a.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. React: Synergizing reasoning and acting in language models. *arXiv preprint arXiv:2210.03629*, 2022b.

Haoxuan You, Haotian Zhang, Zhe Gan, Xianzhi Du, Bowen Zhang, Zirui Wang, Liangliang Cao, Shih-Fu Chang, and Yinfei Yang. Ferret: Refer and ground anything anywhere at any granularity. *arXiv preprint arXiv:2310.07704*, 2023.

Keen You, Haotian Zhang, Eldon Schoop, Floris Weers, Amanda Swearngin, Jeffrey Nichols, Yinfei Yang, and Zhe Gan. Ferret-ui: Grounded mobile ui understanding with multimodal llms. *arXiv e-prints*, pp. arXiv–2404, 2024.

Chaoyun Zhang, Liqun Li, Shilin He, Xu Zhang, Bo Qiao, Si Qin, Minghua Ma, Yu Kang, Qingwei Lin, Saravan Rajmohan, et al. Ufo: A ui-focused agent for windows os interaction. *arXiv preprint arXiv:2402.07939*, 2024a.

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

Hao Zhang, Hongyang Li, Feng Li, Tianhe Ren, Xueyan Zou, Shilong Liu, Shijia Huang, Jianfeng Gao, Lei Zhang, Chunyuan Li, et al. Llava-grounding: Grounded visual chat with large multimodal models. *arXiv e-prints*, pp. arXiv–2312, 2023b.

Haotian Zhang, Haoxuan You, Philipp Dufter, Bowen Zhang, Chen Chen, Hong-You Chen, Tsu-Jui Fu, William Yang Wang, Shih-Fu Chang, Zhe Gan, et al. Ferret-v2: An improved baseline for referring and grounding with large language models. *arXiv preprint arXiv:2404.07973*, 2024b.

Jianguo Zhang, Tian Lan, Ming Zhu, Zuxin Liu, Thai Hoang, Shirley Kokane, Weiran Yao, Juntao Tan, Akshara Prabhakar, Haolin Chen, et al. xlam: A family of large action models to empower ai agent systems. *arXiv preprint arXiv:2409.03215*, 2024c.

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

Shilong Zhang, Peize Sun, Shoufa Chen, Min Xiao, Wenqi Shao, Wenwei Zhang, Yu Liu, Kai Chen, and Ping Luo. Gpt4roi: Instruction tuning large language model on region-of-interest. *arXiv preprint arXiv:2307.03601*, 2023c.

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

Longtao Zheng, Rundong Wang, Xinrun Wang, and Bo An. Synapse: Trajectory-as-exemplar prompting with memory for computer control. In *The Twelfth International Conference on Learning Representations*, 2023.

Shuyan Zhou, Frank F Xu, Hao Zhu, Xuhui Zhou, Robert Lo, Abishek Sridhar, Xianyi Cheng, Tianyue Ou, Yonatan Bisk, Daniel Fried, et al. Webarena: A realistic web environment for building autonomous agents. *arXiv preprint arXiv:2307.13854*, 2023.

Xueyan Zou, Jianwei Yang, Hao Zhang, Feng Li, Linjie Li, Jianfeng Wang, Lijuan Wang, Jianfeng Gao, and Yong Jae Lee. Segment everything everywhere all at once. In *Proceedings of the 37th International Conference on Neural Information Processing Systems*, pp. 19769–19782, 2023.