UI-PRO: A HIDDEN RECIPE FOR BUILDING VISION-LANGUAGE MODELS FOR GUI GROUNDING

Anonymous authors
Paper under double-blind review

ABSTRACT

Building autonomous UI agents that automate user interactions with interfaces has long been a vision in the field of artificial intelligence. Central to these agents is the capability for UI element grounding, which involves accurately locating UI elements (e.g., buttons and links) based on referring expression, such as user intents and functionality descriptions. Developing these agents with robust grounding capabilities using vision-language models (VLMs) offers a promising path forward. However, a practical framework for creating VLMs with strong element grounding capabilities remains under-explored. To address this gap, we conduct systematic experiments within the design space of VLMs to uncover an effective recipe for building VLMs with strong UI element grounding ability. Firstly, we find that fine-tuning with general visual grounding tasks as a warming-up step mitigates the challenges of fine-tuning with downstream UI element grounding data. Next, we explore different fine-tuning sequences of UI grounding training data from various sources and find that a simple-to-complex fine-tuning curriculum can maximize data utility. Moreover, we find that scaling up the size of either the warming-up data or the UI grounding data in downstream fine-tuning significantly enhances UI element grounding accuracy. Lastly, we explore various image feature compression techniques and find that using a convolution-based compressor to compress UI sub-image features significantly enhances the grounding capabilities on high-resolution UI images. Integrating these insights, we successfully develop UI-Pro, an expert VLM that achieves state-of-the-art UI grounding accuracy with fewer parameters across multiple benchmarks. We hope this work serves as a valuable roadmap for researchers in the UI-VLM domain and inspires future research.

1 INTRODUCTION

The concept of autonomous UI agents capable of clicking, typing, and scrolling on behalf of humans as personal assistants is an enticing prospect, as illustrated in Fig. 1. Imagine a UI agent navigating the Internet to perform daily tasks such as using search engines and managing emails, as well as more complex activities like comparing prices across e-commerce platforms and collecting the latest news on stock markets.

At the core of autonomous UI agents is *UI element grounding*, which involves recognizing and locating elements associated with referring expressions. These elements serve as the fundamental building blocks that carry UI functionalities. Accurate grounding allows UI agents to interact effectively with UI components such as buttons, text fields, and images, enabling them to perform tasks like clicking, filling out forms, and extracting information according to user instructions. Developing these agents based on vision-language models (VLMs) (Yin et al., 2023) offers a promising pathway toward realizing this vision. A VLM-based UI agent can directly perceive and interact with UIs as humans do, provided the agent possesses vision and comprehension capabilities that align with those of humans. Although a few prior studies, such as SeeClick (Cheng et al., 2024) and CogAgent (Hong et al., 2023) have explored developing UI-related VLMs, a practical recipe for building VLMs with robust UI grounding capabilities from scratch remains under-explored. Specifically, it is unclear which combination of data is most effective for instilling UI grounding capabilities in VLMs and whether non-UI-related multimodal understanding tasks can serve as useful training data. Furthermore, the optimal design of model architectures and training procedures to enhance the models' ability to perform element grounding on high-resolution UI screenshots is still uncertain. Addressing

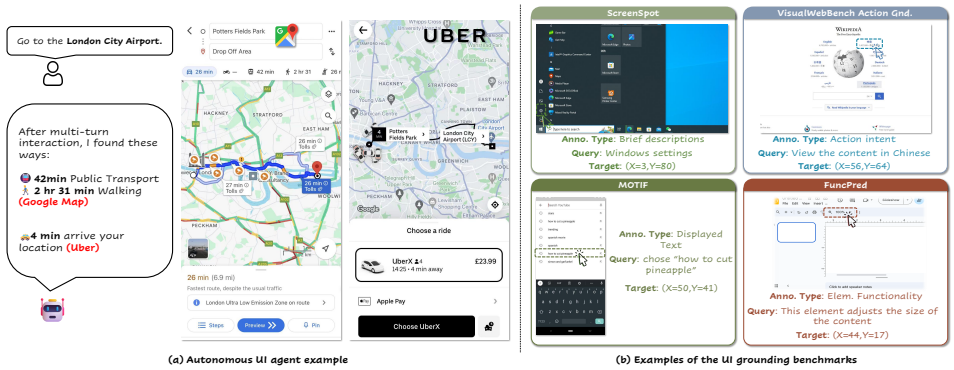

Figure 1: Examples of an autonomous UI agent (left) and existing benchmarks evaluating UI grounding performance (right).

these questions necessitates thorough exploration within the vast design space of VLMs. While several studies (Tong et al., 2024; McKinzie et al., 2024; Laurençon et al., 2024) have examined the significance of various model components and data choices, they predominantly focus on visual question answering (VQA) in natural images, overlooking the complex challenges presented by UI grounding scenarios.

This paper aims to bridge the gap in existing research by providing a practical framework for building VLMs capable of UI grounding. Drawing on pioneering research (Tong et al., 2024; McKinzie et al., 2024; Baechler et al., 2024), we pinpoint three key areas where different studies make distinct design choices: (a) model architecture, particularly vision-language connection modules that enhance the accuracy of locating small elements within high-resolution UI screenshots, presenting new challenges rarely addressed in visual grounding for natural images; (b) training data curation; and (c) fine-tuning procedures. We systematically compare various design choices in a controlled setting to derive empirical insights for each area.

Our findings reveal that: (a) warming up VLMs with general visual grounding task data is essential before fine-tuning the models on downstream UI grounding tasks; (b) organizing UI grounding training data in a simple-to-complex curriculum significantly maximizes data utility through multi-stage fine-tuning; (c) increasing the sizes of both the warming-up data and UI grounding data results in substantial performance gains, well beyond saturation; and (d) a lightweight convolution-based connector is effective for compressing visual features of high-resolution UI images, enabling processing at the original ratio.

Our work distinguishes itself from previous studies (Yao et al., 2022; Cheng et al., 2024; Hong et al., 2023; You et al., 2024b) on UI-oriented VLMs by exploring previously unexamined areas, including warming-up data selection, multi-stage training methodologies, data scaling effects, and UI-oriented connector design. Our findings provide a hidden recipe for building powerful UI VLMs from scratch, circumventing reliance on fine-tuning open-source models.

Based on these insights, we have trained UI-Pro, a VLM with 2.8 billion parameters. UI-Pro demonstrates exceptional performance across multiple UI grounding benchmarks, including grounding by action intents, element appearance, and complex functionality descriptions. Notably, UI-Pro matches the performance of previous UI-oriented models that are nine times its size. We hope our work will benefit the research community and accelerate advancements in UI autonomous agents.

2 Preliminaries

2.1 UI Element Grounding Task

UI element grounding is to locate visual elements within UIs given element annotations. These annotations can be brief, including details such as element appearance, location, and displayed text, or complex, encompassing contextual functionality and action intents, as shown in Fig. 1.

Several element grounding benchmarks have been established for research purposes (Fig 1). **ScreenSpot** (Cheng et al., 2024) is a benchmark for mobile, desktop, and web scenarios, requiring models to locate elements based on brief descriptions. **RefExp** (Bai et al., 2021) focuses on locating elements on mobile devices using crowd-sourced referring expressions. **VisualWebBench** (Liu et al., 2024c) evaluates VLMs in content-rich web environments. In contrast to these benchmarks, **AutoGUI Test (FuncPred)** features complex tasks that require models to locate elements specified by context-specific functionality descriptions. For all these benchmarks, we report the grounding accuracy (%): $\text{Acc } = \sum_{i=1}^{N} \mathbf{1}\left(\text{pred}_i \text{ inside GT bbox }_i\right) / N \times 100$

where $\mathbf{1}$ is an indicator function and N is the number of test samples. This formula denotes the percentage of samples for which the predicted points lie within the bounding boxes of the elements.

Unlike visual grounding aimed at natural scenes (Yu et al., 2016), UI element grounding introduces distinct challenges: 1) High resolution: UIs are typically rendered as high-resolution images, necessitating models that can process large visual inputs effectively. 2) Fine-grained comprehension: UIs often display numerous small elements, which occupy significantly less area than objects in datasets like RefCOCO (Yu et al., 2016) and Visual Genome (VG) (Krishna et al., 2016), requiring enhanced visual understanding to distinguish between highly similar elements. 3) Data insufficiency. Due to the substantial cost of human annotation, the scale of existing open-source datasets for UI understanding is significantly lower than natural image datasets such as COCO (Lin et al., 2014) and LAION-5B (Schuhmann et al., 2022).

2.2 Basics of VLMs

We adopt the popular architecture used by recent VLMs, such as LLaVA (Liu et al., 2023) and Qwen-VL (Bai et al., 2023). These architectures typically combine a pre-trained visual backbone f_ϕ (e.g., ViT (Dosovitskiy et al., 2021)) and a large language model (e.g., Llama Touvron et al. (2023)) to build a model capable of processing both textual and visual inputs. Formally. the visual backbone maps an input image $\mathbf{I} \in \mathbb{R}^{H \times W \times 3}$ to an L-length sequence of patch features $V_{img} \in \mathbb{R}^{L \times h}$ that are then projected into the embedding space of the LLM. Subsequently, the projected visual features $E_{img} \in \mathbb{R}^{L \times D}$ are concatenated with the S-length textual embeddings $E_{txt} \in \mathbb{R}^{S \times D}$ before being fed to the LLM for response generation. The generation process can be formulated as $o = \text{LLM}_\theta([Proj_\omega(f_\phi(\mathbf{I})), Embed(\boldsymbol{t})]$; where $\boldsymbol{t}$, $Proj$, and $Embed$ denotes the text prompt, the vision-language projector, and the embedding module in the LLM, respectively. Given a training sample ($\mathbf{I}$, $\boldsymbol{t}$, o), the VLM is optimized by minimizing the loss $L(\phi, \omega, \theta) = -\log p(o|\mathbf{I}, \boldsymbol{t})$ via gradient descent.

In this paper, we aim to explore the intricate design space of VLMs to develop a comprehensive recipe for building VLMs capable of UI element grounding.

2.3 Downstream UI Element Grounding Training Datasets

To fulfill our aim, two training datasets are utilized as the sources for downstream fine-tuning:

SeeClick (Cheng et al., 2024) provides a dataset that integrates existing tasks in UI element grounding, captioning, and summarization. It comprises two parts: (a) a web portion containing element text grounding and OCR tasks derived from 300k web pages in the latest Common Crawl repository[1]; and (b) a mobile device portion that includes element grounding and captioning tasks generated by applying instruction-following templates to the Widget Captioning and RICO annotations. As shown in Fig. 1, this dataset mainly comprises brief element annotations.

AutoGUI (AutoGUITeam, 2024) is introduced to complement SeeClick. AutoGUI contains 625k UI functionality grounding and captioning tasks that require VLMs to grasp the functional semantics of various UI elements. This dataset is collected on multi-resolution and multi-device screenshots across diverse data domains, providing detailed element functionality annotations related to UI contexts (see Fig. 1). Given that the functionality annotations in this AutoGUI dataset are more detailed and longer than those in SeeClick and that associating these annotations with unique elements among hundreds of counterparts is challenging, this dataset is expected to enhance VLMs' UI element grounding capabilities, albeit with increased complexity.

[1] https://index.commoncrawl.org/

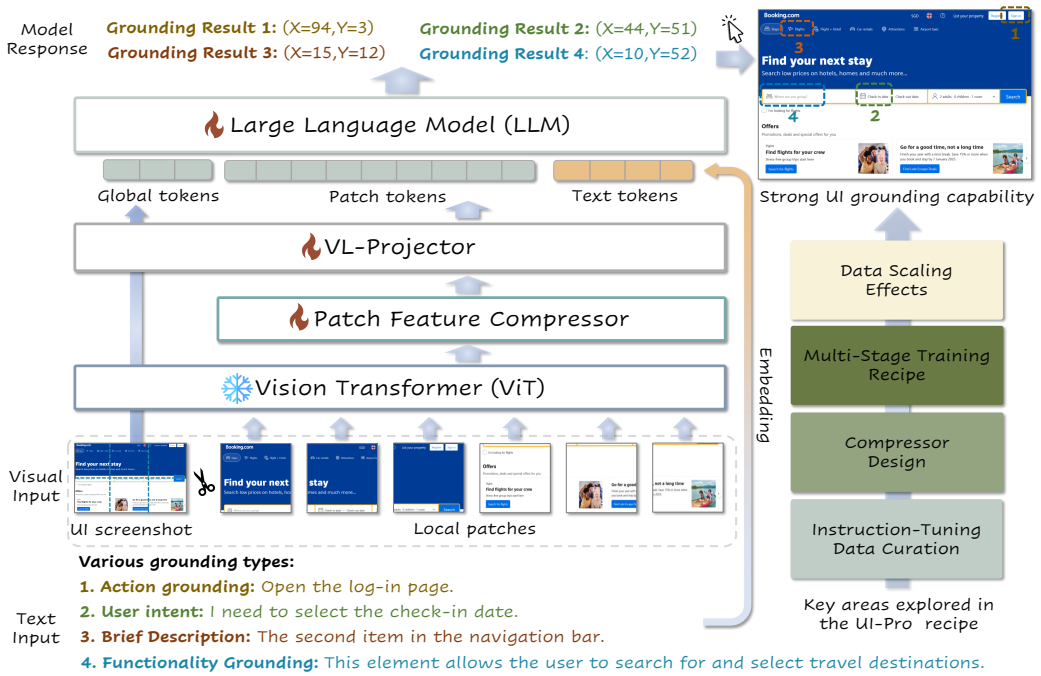

Figure 2: The UI-Pro pipeline and training recipe. The recipe includes (a) Data Curation, (b) Compressor Design, (c) Training Recipe, and (d) Data Scaling for enhancing UI grounding capabilities, as explored in this work. Extensive experiments (conducted in this paper) demonstrate that constructing datasets for warm-up phases, curriculum-based training, scaling datasets, and employing appropriate compression modules play crucial roles in enhancing UI-element grounding performance.

3 Exploring the Design Space of VLMs for Improved UI Grounding

This section investigates design choices related to instruction-tuning data, training procedures, and patch feature compression techniques for UI VLMs. A LLaVA model (Liu et al., 2023) with a pre-trained vision-language projector is utilized as the base model. To process high-resolution UI screenshots, a parameter-free image division strategy (Ye et al., 2023a; Zhang et al., 2024b) is employed to crop the shape-variable screenshots into fixed-size image patches. Please see Fig. 2 for the full pipeline. More implementation details are listed in the Appendix.

3.1 Which Data Type Can be Used to Warm up VLMs?

Our preliminary studies found that directly fine-tuning the base VLM using UI element grounding data resulted in poor grounding accuracy. We hypothesize that the base model needs warming up to adapt to UI element grounding tasks, which require models to output precise numerical coordinates based on cluttered UI screenshots.

To explore suitable warming-up data sources, multiple instruction-tuning tasks shown in Fig. 3 are collected for comparison:

Visual Grounding on Natural Images requires VLMs to output the bounding boxes of target objects given the object descriptions. We convert the bounding box annotations in RefCOCO (Yu et al., 2016), RefCOCO+ (Yu et al., 2016), RefCOCOg Nagaraja et al. (2016), and VG (Krishna et al., 2016) into visual grounding and referring tasks by applying instruction-tuning templates, resulting in 5.7M samples in total.

Question Answering on Natural Images involves generating natural language responses by following text instructions, without coordinate outputs. We utilize the VQA subset, including LLaVA-Pretrain (Liu et al., 2023), COCO Lin et al. (2014), and SAM (Kirillov et al., 2023), of the ShareGPT4V-SFT dataset (Chen et al., 2023a), resulting in 530k samples in total.

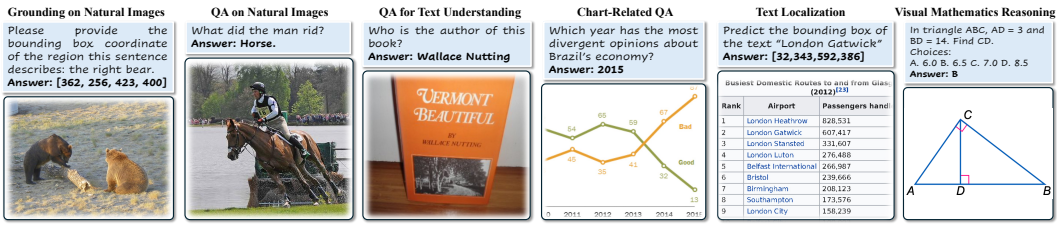

Figure 3: Examples of the task types tested and compared in the warming-up stage.

Table 1: **Evaluating the base model warmed up with various tasks.** The base model is first fine-tuned using a warming-up task and then trained on the downstream AutoGUI task. We can see that visual grounding on natural images significantly enhances accuracy grounding accuracy. Text localization also yields high accuracy. Text-QA, Chart-related QA, and Visual mathematical reasoning tasks provide minimal benefit. VQA on natural images is also not pretty useful. Although ShareGPT4V-SFT includes a variety of tasks, it performs worse than using either pure natural image grounding or text localization task data. Notably, directly fine-tuning with AutoGUI data without warming up results in inferior performance.

Warming-up Task	FuncPred	ScreenSpot	MOTIF	RefExp	VWB EG	VWB AG
Gnd. on Natural Images	42.6	**19.4**	**28.3**	**11.5**	**8.5**	**8.7**
QA on Natural Images	40.2	12.8	19.8	9.6	5.3	1.9
Text Localization	**46.3**	12.5	24.3	8.1	4.8	2.9
Text-QA	39.0	10.4	21.1	9.2	3.6	1.0
Chart-Related QA	36.2	12.3	16.4	10.8	3.6	1.9
Visual Math. Reasoning	37.0	7.9	15.2	5.7	0.7	0.0
ShareGPT4V SFT	35.7	7.5	9.2	7.6	5.1	3.9
None	35.2	4.2	9.6	1.2	1.7	1.0

Question Answering for Text Understanding (Text-QA) focuses on recognizing and interpreting textual contents in images to answer questions. We extract the combination of the TextVQA (Singh et al., 2019), ShareTextQA, and OCR-VQA (Mishra et al., 2019) portions from the ShareGPT4V-SFT dataset, resulting in 102k samples in total.

Text Localization combines text grounding and recognition tasks, requiring the prediction of bounding boxes for situated texts and their recognition. We use the 1M text localization subset of the DocStruct4M data proposed by mPLUG-DocOwl-1.5 (Hu et al., 2024).

Visual Mathematics Reasoning requires VLMs to understand complex mathematical diagrams and formulas for multi-modal reasoning. We combine Inter-GPS (Geometry Problem Solving) (Lu et al., 2021), GeoQA Chen et al. (2021), and MATH-Vision (Wang et al., 2024a), obtaining 82.5k samples.

Chart-Related Question Answering tasks VLMs with answering questions about data visualizations, e.g. scientific diagrams and statistical tables from textbooks and academic papers, challenging visual and logical reasoning over charts. We collect data from ArXivQA Li et al. (2024a), ChartQA (Masry et al., 2022), ScienceQA (Lu et al., 2022), TabMWP (Lu et al., 2023), TextbookQA (Kembhavi et al., 2017), AI2D (Kembhavi et al., 2016), and DVQA (Kafle et al., 2018), curating 394k samples in total.

ShareGPT4V SFT is also used to explore whether combining various types is beneficial. This dataset contains 665k samples, including VQA, visual grounding, and Text-QA tasks.
We restrict the number of samples to 355k by randomly sampling from tasks with more than 355k and resampling those with fewer. This experiment follows a two-stage fine-tuning approach: the base model is first fine-tuned with warming-up tasks, followed by fine-tuning with 125k samples from the AutoGUI dataset. Tab. 1 shows that visual grounding on natural images as the warming-up task significantly enhances accuracy accuracy on the UI grounding benchmarks. Text localization also achieves high accuracy on FuncPred but performs poorly on ScreenSpot. Although text QA, chart-related QA, and visual mathematical reasoning tasks are aimed at enhancing the fine-grained understanding capabilities of VLMs, their overall gains are limited. Interestingly, ShareGPT4V-SFT, which includes diverse tasks, does not provide benefits in the downstream fine-tuning stage for the

Table 2: **Experiments on the fine-tuning curriculum of the warming-up, SeeClick, and AutoGUI datasets.** The fine-tuning procedure is divided into three stages, each of which fine-tuning the base model with a different dataset. Notably, fine-tuning with SeeClick containing simple UI grounding tasks and then with AutoGUI containing complex functionality grounding tasks contributes to high accuracy over the two benchmarks (row 6). Reversing or mixing the two UI datasets leads to deteriorated performance on FuncPred (rows 7 and 8). These results indicate that organizing the two UI grounding datasets with a simple-to-complex curriculum helps to maximize data utility.

No.	SFT-1	SFT-2	SFT-3	FuncPred	ScreenSpot
r1	-	SeeClick	-	17.3	39.9
r2	-	-	AutoGUI	46.3	14.3
r3	-	SeeClick	AutoGUI	56.7	41.1
r4	Gnd.	SeeClick	-	20.8	44.0
r5	Gnd.	-	AutoGUI	52.0	24.9
r6	Gnd.	SeeClick	AutoGUI	**57.7**	**44.7**
r7	Gnd.	AutoGUI	SeeClick	22.3	43.9
r8	Gnd.	-	SeeClick+AutoGUI	52.0	44.4

UI dataset, suggesting that this diverse dataset is not an ideal warming-up source for enhancing UI element grounding capabilities, despite its common use in existing VLM studies (McKinzie et al., 2024; Tong et al., 2024). In summary, these results demonstrate that grounding tasks from either the natural image or text-rich scenarios serve as desirable warming-up data sources.

***Finding* 1.** Directly fine-tuning VLMs with UI element grounding data can lead to training difficulty. Utilizing visual grounding tasks from either natural or text-rich scenarios as a warming-up step significantly enhances downstream fine-tuning with UI element grounding tasks.

3.2 What Fine-tuning Curriculum Can Maximize Dataset Utility?

Given a warming-up dataset, a UI grounding dataset with simple annotations (SeeClick), and one with complex functionality semantics (AutoGUI), a question arises of how we can optimize the fine-tuning order to fully leverage the advantages of these datasets. In this experiment, we aim to find a suitable fine-tuning procedure for utilizing datasets from various sources to enhance VLMs' UI element grounding capability. We explore different fine-tuning orders of the three datasets and compare the performances. Specifically, the fine-tuning process is divided into three stages: the first stage uses visual grounding on natural images (355k samples) to warm up the base model according to the finding in Sec. 3.1; the second uses 355k samples from the simple tasks in SeeClick; the third uses the 625k complex tasks in AutoGUI. Each stage is run for one epoch.

The results in Tab. 2 demonstrate that initial fine-tuning with the warming-up task consistently enhances downstream fine-tuning with UI grounding data (r4 vs. r1, r5 vs. r2, and r6 vs. r3), especially when the downstream task is the hard functionality grounding task of AutoGUI. This observation aligns with findings in Sec. 3.1. Fine-tuning with first SeeClick (simple) and then AutoGUI (complex) generally yields better performance than variants that fine-tune exclusively with either SeeClick or AutoGUI (r3 vs. r1 and r2). This trend persists even when models are warmed up, obtaining accuracy gains of 5.7 and 18.8 on the FuncPred and ScreenSpot, respectively (r6 vs. r5). An exception appears when comparing r6 and r4, where performance on ScreenSpot remains unchanged despite a significant increase on FuncPred, likely due to a small domain gap between the functionality grounding task of AutoGUI and the grounding-by-brief-descriptions task of the ScreenSpot benchmark.

Interestingly, reversing the order of SeeClick and AutoGUI (r7 vs. r6) leads to a significant performance drop of 35.4 on AutoGUI. Additionally, mixing these tasks for fine-tuning leads to a decrease in FuncPred (r8 vs. r6). These findings indicate that the base model requires warming up with the simpler UI grounding task before fine-tuning with more complex tasks. In summary, these results indicate that organizing the three datasets in a simple-to-complex curriculum maximizes data utility.

***Finding* 2.** In the downstream fine-tuning stage after warming up, structuring UI grounding task datasets in a simple-to-complex curriculum significantly enhances data utility.

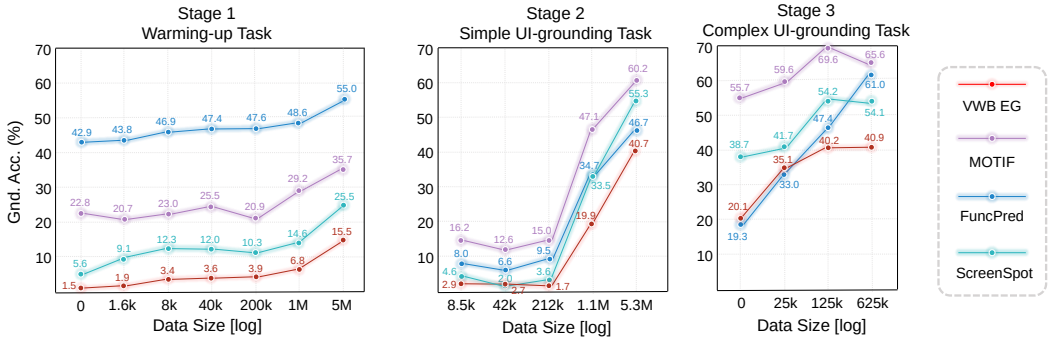

Figure 4: **Scaling effects for the warming up, simple UI grounding (SeeClick), and complex UI grounding (AutoGUI) data.** This figure highlights that increasing the warming-up and UI grounding data remarkably improves performance. Scaling the AutoGUI data shows modest gains, with peak performance observed at 125k samples for MOTIF and ScreenSpot, suggesting potential overfitting.

3.3 What is the Significance of Fine-Tuning Data Size?

Exploring the effects of data size scaling in training VLMs is pivotal for optimizing performance as increased data has been observed to lead to enhanced model generalization and performance (Kaplan et al., 2020; Brown et al., 2020; Zhao et al., 2023; Liu et al., 2024a; Karamcheti et al., 2024).

We systematically assess how varying fine-tuning data sizes impact the performance of VLMs in the UI element grounding tasks. Following the insight in Sec. 3.2, we adopt a three-stage fine-tuning procedure, employing the warming up, SeeClick, and AutoGUI data in stages 1, 2, and 3, respectively.

Scaling warming-up data in stage 1. Inspired by the power-law scaling observed in (Kaplan et al., 2020), we scale the visual grounding on natural image data, as discussed in Sec. 3.1, across seven levels: 0, 1.6k, 8k, 40k, 200k, 1M, and 5M samples. 212k samples of SeeClick data are used in stage 2 and 625k AutoGUI data are used in stage 3.

Scaling simple UI-grounding data in stage 2. The SeeClick training data is scaled across five levels: 8.5k, 42k, 212k, 1.06Mk, and 5.3M samples. 5M samples of the warming-up data are used in stage 1 and 625k AutoGUI data are used in stage 3.

Scaling complex UI-grounding data in stage 3. The AutoGUI training data is extracted and scaled across four levels: 0, 25k, 125k, and 625k samples. This experiment utilizes 5M samples of the warming-up data and 212k samples of SeeClick data.

The results in Fig. 4 show that scaling up the warming-up data in stage 1 contributes to significant improvements in benchmark performance, even though the domain of this data (natural images) differs from the UI-specific data used in subsequent fine-tuning stages. This suggests that the model acquires a preliminary capability of fine-grained spatial localization and numerical coordinate generation, which are essential for tackling the more challenging UI grounding tasks downstream. Scaling up SeeClick data in stage 2 also brings significant performance gains across all the benchmarks, with the scale of 212k serving as a critical reflection point. Scaling the AutoGUI data in stage 3 results in modest improvements, peaking at 125k on MOTIF and ScreenSpot. The FuncPred accuracy continues to rise, likely due to its alignment with the AutoGUI task domain, while performance on the other benchmarks plateaus or slightly declines, possibly indicating overfitting to the AutoGUI task format. In summary, while scaling the warming-up and UI grounding data enhances performance, attention should be paid to the risk of overfitting.

***Finding* 3.** Scaling warming-up data and UI grounding data significantly enhances element grounding accuracy. It is also important to remain cautious of potential overfitting when finetuning with complex UI grounding data.

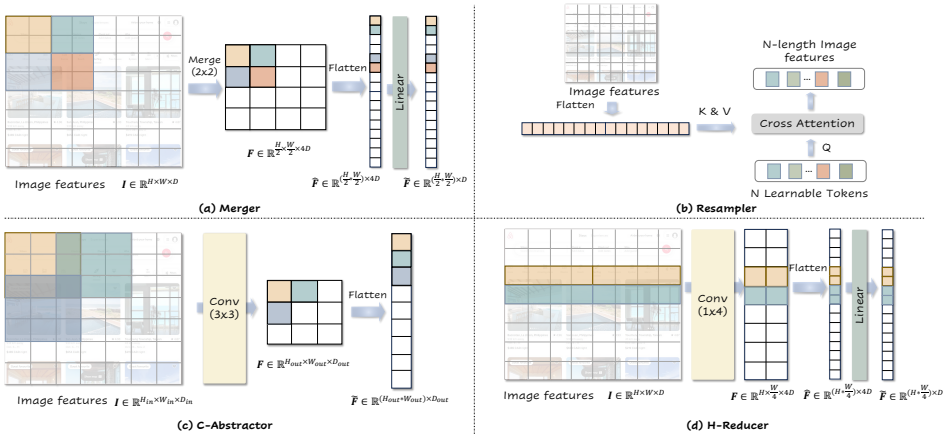

Figure 5: **Comparing the patch feature compressors.** (a) Merger concatenates the nearby 4 tokens into a new token along the channel dimension and then reduces the channel dimension to 1/4 with an MLP. (b) Resampler uses a fixed set of learnable latent queries that interact with each patch feature through cross-attention, outputting a fixed-length visual feature. (c) C-Abstractor employs a traditional convolution network to compress patch features. (d) H-Reducer uses a convolution network whose kernel size and stride size are set as 1×4 to fuse horizontal 4 visual tokens.

3.4 Which Patch Feature Compression Approach Is Beneficial?

UIs captured at super-high resolutions result in thousands of visual tokens when processed with the division technique. For example, a 720p screenshot will be converted into 4608 tokens and a 4kHD will exceed 48k. These extensive features may possess redundant details that interfere with VLM inference and cause an unbearably high computational budget.

To build a UI VLM capable of efficiently processing high-resolution screenshots, we explore various designs of the patch feature compressor, as shown in Fig. 5: (a) **Merging compressor (Merger)**(Ye et al., 2023a): This compressor concatenates $N = 4$ adjacent tokens in square regions along the channel dimension andcompresses the concatenated features using a single-layer MLP. (b) **Resampler** (Alayrac et al., 2022): This compressor reduces visual features to a fixed number of tokens by utilizing a set of learnable queries to cross-attend to the visual features. (c) **Convolution-based compressor**: This compressor reshapes the visual features to align them with image dimensionality, processes them through a convolutional network, and flattens them back into visual tokens. Apart from convolution with square-shaped kernels, i.e., **C-Abstractor** (Cha et al., 2024), we also test the **H-reducer** (Dong et al., 2024), a type with stripe-shaped kernels (1×4), which is tailored for horizontal text layouts in document understanding scenarios. (d) **MLP**: Directly using an MLP to process the patch features without compression.

This experiment uses 1M warming-up samples, 1.1M SeeClick data, and 125k AutoGUI data to fine-tune the base model for one epoch. To ensure a fair comparison, the numbers of parameters of these compressors are roughly equalized by adjusting their hyper-parameters.

The results in Tab. 3 show that the two convolution-based compressors achieve superior grounding accuracy, with C-Abstractor leading on five benchmarks. Although H-Reducer is tailored for document inputs, it is inferior to C-Abstractor which uses square-shaped kernels, probably because the UI screenshots display a higher proportion of flexibly arranged visual contents (i.e., icons and images), compared to text-rich documents. Resampler performs poorly as its cross-attention mechanism possibly leads to a loss of spatial information, which is crucial for grounding tasks that require precise annotation-region associations (Cha et al., 2024). In contrast, Merger simply fuses nearby

Table 3: **Comparing the performances of introducing different patch feature compressors.** The convolution-based compressor obtains higher accuracy than the other types, with C-Abstractor exhibiting leading performances on five benchmarks. Without a compressor, the model can only employ the low-resolution raw image, resulting in significantly inferior performance.

Compressor Type	FuncPred	ScreenSpot	MOTIF	RefExp	VWB EG	VWB AG
None	20.1	3.8	18.0	4.2	1.5	1.0
MLP (Liu et al., 2024b)	26.2	13.6	36.1	20.0	7.0	20.4
Merger (Ye et al., 2023a)	35.9	38.5	51.3	26.0	16.0	26.2
Resampler (Alayrac et al., 2022)	35.1	29.2	50.3	21.6	13.6	13.6
C-Abstractor (Cha et al., 2024)	34.3	**42.1**	**58.0**	**32.0**	**16.5**	**30.1**
H-Reducer (Dong et al., 2024)	**37.0**	39.7	55.3	28.3	10.2	27.2

Table 4: **Comparing UI-Pro with leading VLMs on the UI element grounding benchmarks.** The results demonstrate that UI-Pro achieves impressive grounding accuracy with much fewer parameters. AnyRes means that the method uses an image division strategy to handle images with variable resolutions.

Model	Size	Input Res.	FuncPred	ScreenSpot	MOTIF	RefExp	VWB EG	VWB AG
LLaVA-1.5 (Liu et al., 2024a)	7B	336	3.2	5.0	7.2	4.2	12.1	13.6
LLaVA-1.5 (Liu et al., 2024a)	13B	336	5.8	11.2	12.3	20.3	16.7	9.7
LLaVA-1.6 (Liu et al., 2024b)	34B	AnyRes	4.4	10.3	7.0	29.1	19.9	17.0
SliME (Zhang et al., 2024b)	8B	AnyRes	3.2	13.0	7.0	8.3	6.1	4.9
MiniCPM-V-2.6 (Yao et al., 2024)	8B	AnyRes	16.5	33.0	12.9	29.3	9.4	21.7
Qwen-VL (Bai et al., 2023)	10B	448	3.0	5.2	7.8	8.0	1.7	3.9
Qwen2-VL (Wang et al., 2024b)	7B	AnyRes	7.8	26.1	16.7	32.4	3.9	3.9
CogAgent (Hong et al., 2023)	18B	1120	29.3	47.4	46.7	35.0	**55.7**	**59.2**
SeeClick (Cheng et al., 2024)	10B	448	19.8	53.4	11.1	**58.1**	39.2	27.2
UI-Pro-Gemma-2B (ours)	2.8B	AnyRes	**46.3**	**56.3**	**64.3**	44.6	43.8	33.0

four tokens, surpassing Resampler by a large margin. The MLP-based variant and the one without compression both show weak UI grounding ability, suggesting the necessity of a compression module. In summary, these results suggest that convolution-based compressors are suitable for enhancing VLMs' UI grounding capabilities.

***Finding* 4.** Employing compression modules to reduce patch features is crucial for VLMs that utilize an image division strategy. The use of convolutional networks to compress local patch features significantly enhances the UI grounding capabilities of VLMs.

4 State of the Art Performance of UI-Pro

Finally, we leverage the findings from the previous experiments to build UI-Pro. We train UI-Pro using Gemma-1.1-2B (Team et al., 2024) and LLaMA-3.2-Instruct-3B AI@Meta (2024) as the base LLM and OpenAI CLIP ViT-L/14@336 (Dosovitskiy et al., 2021) as the visual encoder. The training process begins with 5M samples of visual grounding on natural images for warming up, followed by fine-tuning with 5.3M SeeClick and 125k AutoGUI samples, adhering to a simple-to-complex curriculum. We utilize C-Abstractor as the patch feature compressor.

Tab. 4 show that compared with existing VLMs, UI-Pro exhibits impressively high accuracy on the UI element grounding benchmarks. Notably, it achieves this with only one-fifth the model size of CogAgent, a leading UI-oriented VLM, surpassing it across five benchmarks and establishing a new state-of-the-art. In contrast, VLMs primarily designed for universal multimodal comprehension, such as Qwen-VL, MiniCPM-V-2.6, and LLaVA-1.6, struggle with UI element grounding tasks, highlighting a potential disconnect between their design strategies and the complexities of UI grounding scenarios. Overall, with our four key findings, we can build VLMs that possess strong UI element grounding capabilities.

5 Related Works

5.1 Recent Advancement of VLMs

There has been a significant rise in reasearch focused on improving LLMs by integrating both visual and textual data (Alayrac et al., 2022; Chen et al., 2023b; Li et al., 2023; Lin et al., 2023a; Liu et al., 2023; Lin et al., 2023b; Chen et al., 2023c; Lu et al., 2024; Bai et al., 2023; Wang et al., 2024b; Zhu et al., 2024; Wang et al., 2024c; Li et al., 2024b; Zhang et al., 2024a; You et al., 2024a; Laurençon et al., 2024; Peng et al., 2024; Driess et al., 2023), which has led to the development of VLMs. Flamingo (Alayrac et al., 2022), utilizes combined visual and language prompts and has demonstrated exceptional few-shot visual question-answering abilities. With the advancements brought by GPT-4 (Team, 2024), both academic and industrial efforts have been made to make its multimodal reasoning capabilities more accessible. LLaVA (Liu et al., 2023) and LLaMA-Adapter (Zhang et al., 2024a) have worked to align vision encoders (Dosovitskiy et al., 2021) with LLMs to support visual instruction following. Models like VisionLLM (Wang et al., 2024c), Ferret (You et al., 2024a), and Qwen-VL (Bai et al., 2023) have further developed strong visual grounding abilities. LLaVA-Next Liu et al. (2024b), Monkey Li et al. (2024b), LLaVA-UHD Guo et al. (2024), Qwen2-VL Wang et al. (2024b) enhanced the perception resolution of VLMs. Moreover, research is expanding to VLM applications in contexts with rich textual imagery (Tang et al., 2022; Ye et al., 2023b;a; Liu et al., 2024d) and embodied interactions (Driess et al., 2023; Mu et al., 2023), unlocking new possibilities in multimodal reasoning. Additionally, some works Laurençon et al. (2024); McKinzie et al. (2024) give a comprehensive study on building VLMs, highlighting the impact of various design components and data choices on model performance. Despite these advancements, there has been no systematic approach proposed for data collection, model design, or training frameworks specifically targeting VLMs in UI environments, highlighting a critical gap in the research.

5.2 Existing UI Datasets and Benchmarks

In contrast to well-established natural image datasets (Russakovsky et al., 2014; Schuhmann et al., 2022), datasets focused on UI understanding have received less attention in the field of computer vision. Some efforts have been made to develop datasets for mobile UI modeling (Wang et al., 2021; Li et al., 2020a;b; Bai et al., 2021; Burns et al., 2022), with many of these efforts centered on the RICO dataset (Deka et al., 2017), which contains 72K Android app screenshots. Notable examples include Widget Captioning (Li et al., 2020a), which examines the captions and linguistic features of UI elements, and RICOSCA (Li et al., 2020b), which maps single-step instructions to corresponding UI elements. More recently, MoTIF (Burns et al., 2022) has gained attention alongside the growing interest in web-based scenarios. WebShop (Yao et al., 2022), for instance, was an early effort to introduce a simplified simulator for web navigation tasks. Subsequent projects like Mind2Web (Deng et al., 2024) and WebArena (Zhou et al., 2023) have focused on creating realistic and reproducible web environments to enhance web agent performance. VisualWebBench (Liu et al., 2024c) has also contributed by establishing a robust evaluation framework for VLMs, specifically targeting UI grounding. To address the issue of limited data, recent studies like SeeClick (Cheng et al., 2024) and CogAgent (Hong et al., 2023) have leveraged Common Crawl data to construct large-scale datasets, though these datasets often include noisy HTML code snippets. AITW (Rawles et al., 2023) has been introduced to focus on interpreting high-level instructions in Android environments. Existing UI-VLMs have primarily focused on fine-tuning open-source models using these datasets, but there is still a lack of detailed solutions for effectively enhancing their UI grounding capabilities.

6 Conclusion

This paper introduces a practical framework for building VLMs with strong UI element grounding capability. We identified key strategies, including warming up with visual grounding tasks, employing a simple-to-complex fine-tuning curriculum, and scaling data sizes, all of which significantly optimize grounding accuracy. Our findings on image feature compression further improve grounding accuracy for high-resolution UI images. The integration of these findings resulted in UI-Pro, a state-of-the-art VLM that achieves impressive grounding accuracy with fewer parameters. Hope this research provides a roadmap for future studies in intelligent UI agent development.

Reproducibility Statement

UI-Pro is fully reproducible. The fine-tuning code is based on the open-source LLaVA repo and all the used training data in the experiments are also open-sourced. As the four findings proposed in the paper are easy to put into practice, readers can reproduce our results by modifying LLaVA repo according to the model designs Fig. 5, fine-tuning curriculum in Sec 3.2, and hyperparameter settings in the Appendix.

References

AI@Meta. Llama 3 model card. 2024. URL https://github.com/meta-llama/llama3/blob/main/MODEL_CARD.md.

Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katherine Millican, Malcolm Reynolds, et al. Flamingo: a visual language model for few-shot learning. *Advances in neural information processing systems*, 35:23716–23736, 2022.

AutoGUITeam. Autogui dataset, 2024. URL https://huggingface.co/AutoGUI.

Gilles Baechler, Srinivas Sunkara, Maria Wang, Fedir Zubach, Hassan Mansoor, Vincent Etter, Victor Cărbune, Jason Lin, Jindong Chen, and Abhanshu Sharma. Screenai: A vision-language model for ui and infographics understanding, 2024. URL https://arxiv.org/abs/2402.04615.

Chongyang Bai, Xiaoxue Zang, Ying Xu, Srinivas Sunkara, Abhinav Rastogi, Jindong Chen, and Blaise Agüera y Arcas. Uibert: Learning generic multimodal representations for ui understanding. In *International Joint Conference on Artificial Intelligence*, 2021. URL https://api.semanticscholar.org/CorpusID:236493482.

Jinze Bai, Shuai Bai, Shusheng Yang, Shijie Wang, Sinan Tan, Peng Wang, Junyang Lin, Chang Zhou, and Jingren Zhou. Qwen-vl: A versatile vision-language model for understanding, localization, text reading, and beyond. 2023.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 1877–1901. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper_files/paper/2020/file/1457c0d6bfcb4967418bfb8ac142f64a-Paper.pdf.

Andrea Burns, Deniz Arsan, Sanjna Agrawal, Ranjitha Kumar, Kate Saenko, and Bryan A. Plummer. A dataset for interactive vision-language navigation with unknown command feasibility. In *European Conference on Computer Vision*, 2022. URL https://api.semanticscholar.org/CorpusID:251040563.

Junbum Cha, Wooyoung Kang, Jonghwan Mun, and Byungseok Roh. Honeybee: Locality-enhanced projector for multimodal llm. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 13817–13827, June 2024.

Jiaqi Chen, Jianheng Tang, Jinghui Qin, Xiaodan Liang, Lingbo Liu, Eric Xing, and Liang Lin. GeoQA: A geometric question answering benchmark towards multimodal numerical reasoning. In Chengqing Zong, Fei Xia, Wenjie Li, and Roberto Navigli (eds.), *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pp. 513–523, Online, August 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.findings-acl.46. URL https://aclanthology.org/2021.findings-acl.46.

Lin Chen, Jisong Li, Xiaoyi Dong, Pan Zhang, Conghui He, Jiaqi Wang, Feng Zhao, and Dahua Lin. Sharegpt4v: Improving large multi-modal models with better captions. *arXiv preprint arXiv:2311.12793*, 2023a.

Xi Chen, Xiao Wang, Soravit Changpinyo, AJ Piergiovanni, Piotr Padlewski, Daniel Salz, Sebastian Goodman, Adam Grycner, Basil Mustafa, Lucas Beyer, Alexander Kolesnikov, Joan Puigcerver, Nan Ding, Keran Rong, Hassan Akbari, Gaurav Mishra, Linting Xue, Ashish V Thapliyal, James Bradbury, Weicheng Kuo, Mojtaba Seyedhosseini, Chao Jia, Burcu Karagol Ayan, Carlos Riquelme Ruiz, Andreas Peter Steiner, Anelia Angelova, Xiaohua Zhai, Neil Houlsby, and Radu Soricut. PaLI: A jointly-scaled multilingual language-image model. In *The Eleventh International Conference on Learning Representations*, 2023b. URL https://openreview.net/forum?id=mWVoBz4W0u.

Zhe Chen, Jiannan Wu, Wenhai Wang, Weijie Su, Guo Chen, Sen Xing, Muyan Zhong, Qinglong Zhang, Xizhou Zhu, Lewei Lu, Bin Li, Ping Luo, Tong Lu, Yu Qiao, and Jifeng Dai. Internvl: Scaling up vision foundation models and aligning for generic visual-linguistic tasks. *arXiv preprint arXiv:2312.14238*, 2023c.

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.

Biplab Deka, Zifeng Huang, Chad Franzen, Joshua Hibschman, Daniel Afergan, Yang Li, Jeffrey Nichols, and Ranjitha Kumar. Rico: A mobile app dataset for building data-driven design applications. In *Proceedings of the 30th annual ACM symposium on user interface software and technology*, pp. 845–854, 2017.

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.

Xiaoyi Dong, Pan Zhang, Yuhang Zang, Yuhang Cao, Bin Wang, Linke Ouyang, Songyang Zhang, Haodong Duan, Wenwei Zhang, Yining Li, Hang Yan, Yang Gao, Zhe Chen, Xinyue Zhang, Wei Li, Jingwen Li, Wenhai Wang, Kai Chen, Conghui He, Xingcheng Zhang, Jifeng Dai, Yu Qiao, Dahua Lin, and Jiaqi Wang. Internlm-xcomposer2-4khd: A pioneering large vision-language model handling resolutions from 336 pixels to 4k hd. *arXiv preprint arXiv:2404.06512*, 2024.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=YicbFdNTTy.

Danny Driess, Fei Xia, Mehdi S. M. Sajjadi, Corey Lynch, Aakanksha Chowdhery, Brian Ichter, Ayzaan Wahid, Jonathan Tompson, Quan Vuong, Tianhe Yu, Wenlong Huang, Yevgen Chebotar, Pierre Sermanet, Daniel Duckworth, Sergey Levine, Vincent Vanhoucke, Karol Hausman, Marc Toussaint, Klaus Greff, Andy Zeng, Igor Mordatch, and Pete Florence. Palm-e: An embodied multimodal language model. In *arXiv preprint arXiv:2303.03378*, 2023.

Zonghao Guo, Ruyi Xu, Yuan Yao, Junbo Cui, Zanlin Ni, Chunjiang Ge, Tat-Seng Chua, Zhiyuan Liu, and Gao Huang. LLaVA-UHD: an lmm perceiving any aspect ratio and high-resolution images. In *ECCV*, 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. *arXiv preprint arXiv:2312.08914*, 2023.

Anwen Hu, Haiyang Xu, Jiabo Ye, Ming Yan, Liang Zhang, Bo Zhang, Chen Li, Ji Zhang, Qin Jin, Fei Huang, et al. mplug-docowl 1.5: Unified structure learning for ocr-free document understanding. *arXiv preprint arXiv:2403.12895*, 2024.

Kushal Kafle, Brian Price, Scott Cohen, and Christopher Kanan. Dvqa: Understanding data visualizations via question answering. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5648–5656, 2018. doi: 10.1109/CVPR.2018.00592.

Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B. Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeff Wu, and Dario Amodei. Scaling laws for neural language models. *ArXiv*, abs/2001.08361, 2020. URL https://api.semanticscholar.org/CorpusID:210861095.

Siddharth Karamcheti, Suraj Nair, Ashwin Balakrishna, Percy Liang, Thomas Kollar, and Dorsa Sadigh. Prismatic vlms: Investigating the design space of visually-conditioned language models, 2024. URL https://arxiv.org/abs/2402.07865.

Aniruddha Kembhavi, Mike Salvato, Eric Kolve, Minjoon Seo, Hannaneh Hajishirzi, and Ali Farhadi. A diagram is worth a dozen images. In Bastian Leibe, Jiri Matas, Nicu Sebe, and Max Welling (eds.), *Computer Vision – ECCV 2016*, pp. 235–251, Cham, 2016. Springer International Publishing. ISBN 978-3-319-46493-0.

Aniruddha Kembhavi, Minjoon Seo, Dustin Schwenk, Jonghyun Choi, Ali Farhadi, and Hannaneh Hajishirzi. Are you smarter than a sixth grader? textbook question answering for multimodal machine comprehension. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5376–5384, 2017. doi: 10.1109/CVPR.2017.571.

Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C. Berg, Wan-Yen Lo, Piotr Dollar, and Ross Girshick. Segment anything. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 4015–4026, October 2023.

Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A. Shamma, Michael S. Bernstein, and Fei-Fei Li. Visual genome: Connecting language and vision using crowdsourced dense image annotations, 2016. URL https://arxiv.org/abs/1602.07332.

Hugo Laurençon, Léo Tronchon, Matthieu Cord, and Victor Sanh. What matters when building vision-language models?, 2024.

Bo Li, Yuanhan Zhang, Liangyu Chen, Jinghao Wang, Jingkang Yang, and Ziwei Liu. Otter: A multi-modal model with in-context instruction tuning, 2023.

Lei Li, Yuqi Wang, Runxin Xu, Peiyi Wang, Xiachong Feng, Lingpeng Kong, and Qi Liu. Multimodal ArXiv: A dataset for improving scientific comprehension of large vision-language models. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 14369–14387, Bangkok, Thailand, August 2024a. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-long.775. URL https://aclanthology.org/2024.acl-long.775.

Y. Li, Gang Li, Luheng He, Jingjie Zheng, Hong Li, and Zhiwei Guan. Widget captioning: Generating natural language description for mobile user interface elements. In *Conference on Empirical Methods in Natural Language Processing*, 2020a. URL https://api.semanticscholar.org/CorpusID:222272319.

Yang Li, Jiacong He, Xiaoxia Zhou, Yuan Zhang, and Jason Baldridge. Mapping natural language instructions to mobile ui action sequences. *ArXiv*, abs/2005.03776, 2020b. URL https://api.semanticscholar.org/CorpusID:218571167.

Zhang Li, Biao Yang, Qiang Liu, Zhiyin Ma, Shuo Zhang, Jingxu Yang, Yabo Sun, Yuliang Liu, and Xiang Bai. Monkey: Image resolution and text label are important things for large multi-modal models. In *proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2024b.

Ji Lin, Hongxu Yin, Wei Ping, Yao Lu, Pavlo Molchanov, Andrew Tao, Huizi Mao, Jan Kautz, Mohammad Shoeybi, and Song Han. Vila: On pre-training for visual language models. *ArXiv*, abs/2312.07533, 2023a. URL https://api.semanticscholar.org/CorpusID:266174746.

Tsung-Yi Lin, Michael Maire, Serge J. Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft coco: Common objects in context. In *European Conference on Computer Vision*, 2014. URL https://api.semanticscholar.org/CorpusID:14113767.

Ziyi Lin, Chris Liu, Renrui Zhang, Peng Gao, Longtian Qiu, Han Xiao, Han Qiu, Chen Lin, Wenqi Shao, Keqin Chen, Jiaming Han, Siyuan Huang, Yichi Zhang, Xuming He, Hongsheng Li, and Yu Qiao. Sphinx: The joint mixing of weights, tasks, and visual embeddings for multi-modal large language models, 2023b.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning, 2023.

Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. Improved baselines with visual instruction tuning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 26296–26306, June 2024a.

Haotian Liu, Chunyuan Li, Yuheng Li, Bo Li, Yuanhan Zhang, Sheng Shen, and Yong Jae Lee. Llava-next: Improved reasoning, ocr, and world knowledge, January 2024b. URL https://llava-vl.github.io/blog/2024-01-30-llava-next/.

Junpeng Liu, Yifan Song, Bill Yuchen Lin, Wai Lam, Graham Neubig, Yuanzhi Li, and Xiang Yue. Visualwebbench: How far have multimodal llms evolved in web page understanding and grounding? *arXiv preprint arXiv:2404.05955*, 2024c.

Yuliang Liu, Biao Yang, Qiang Liu, Zhang Li, Zhiyin Ma, Shuo Zhang, and Xiang Bai. Textmonkey: An ocr-free large multimodal model for understanding document. *arXiv preprint arXiv:2403.04473*, 2024d.

Haoyu Lu, Wen Liu, Bo Zhang, Bingxuan Wang, Kai Dong, Bo Liu, Jingxiang Sun, Tongzheng Ren, Zhuoshu Li, Hao Yang, Yaofeng Sun, Chengqi Deng, Hanwei Xu, Zhenda Xie, and Chong Ruan. Deepseek-vl: Towards real-world vision-language understanding, 2024.

Pan Lu, Ran Gong, Shibiao Jiang, Liang Qiu, Siyuan Huang, Xiaodan Liang, and Song-Chun Zhu. Inter-GPS: Interpretable geometry problem solving with formal language and symbolic reasoning. In Chengqing Zong, Fei Xia, Wenjie Li, and Roberto Navigli (eds.), *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pp. 6774–6786, Online, August 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.acl-long.528. URL https://aclanthology.org/2021.acl-long.528.

Pan Lu, Swaroop Mishra, Tanglin Xia, Liang Qiu, Kai-Wei Chang, Song-Chun Zhu, Oyvind Tafjord, Peter Clark, and Ashwin Kalyan. Learn to explain: Multimodal reasoning via thought chains for science question answering. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh (eds.), *Advances in Neural Information Processing Systems*, volume 35, pp. 2507–2521. Curran Associates, Inc., 2022. URL https://proceedings.neurips.cc/paper_files/paper/2022/file/11332b6b6cf4485b84afadb1352d3a9a-Paper-Conference.pdf.

Pan Lu, Liang Qiu, Kai-Wei Chang, Ying Nian Wu, Song-Chun Zhu, Tanmay Rajpurohit, Peter Clark, and Ashwin Kalyan. Dynamic prompt learning via policy gradient for semi-structured mathematical reasoning. In *International Conference on Learning Representations (ICLR)*, 2023.

Ahmed Masry, Xuan Long Do, Jia Qing Tan, Shafiq Joty, and Enamul Hoque. ChartQA: A benchmark for question answering about charts with visual and logical reasoning. In Smaranda Muresan, Preslav Nakov, and Aline Villavicencio (eds.), *Findings of the Association for Computational Linguistics: ACL 2022*, pp. 2263–2279, Dublin, Ireland, May 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.findings-acl.177. URL https://aclanthology.org/2022.findings-acl.177.

Brandon McKinzie, Zhe Gan, Jean-Philippe Fauconnier, Sam Dodge, Bowen Zhang, Philipp Dufter, Dhruti Shah, Xianzhi Du, Futang Peng, Floris Weers, Anton Belyi, Haotian Zhang, Karanjeet Singh, Doug Kang, Ankur Jain, Hongyu Hè, Max Schwarzer, Tom Gunter, Xiang Kong, Aonan

Zhang, Jianyu Wang, Chong Wang, Nan Du, Tao Lei, Sam Wiseman, Guoli Yin, Mark Lee, Zirui Wang, Ruoming Pang, Peter Grasch, Alexander Toshev, and Yinfei Yang. Mm1: Methods, analysis & insights from multimodal llm pre-training, 2024. URL https://arxiv.org/abs/2403.09611.

Anand Mishra, Shashank Shekhar, Ajeet Kumar Singh, and Anirban Chakraborty. Ocr-vqa: Visual question answering by reading text in images. In *2019 International Conference on Document Analysis and Recognition (ICDAR)*, pp. 947–952, 2019. doi: 10.1109/ICDAR.2019.00156.

Yao Mu, Qinglong Zhang, Mengkang Hu, Wenhai Wang, Mingyu Ding, Jun Jin, Bin Wang, Jifeng Dai, Yu Qiao, and Ping Luo. EmbodiedGPT: Vision-language pre-training via embodied chain of thought. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. URL https://openreview.net/forum?id=IL5zJqfxAa.

Varun K. Nagaraja, Vlad I. Morariu, and Larry S. Davis. Modeling context between objects for referring expression understanding. In Bastian Leibe, Jiri Matas, Nicu Sebe, and Max Welling (eds.), *Computer Vision – ECCV 2016*, pp. 792–807, Cham, 2016. Springer International Publishing. ISBN 978-3-319-46493-0.

Zhiliang Peng, Wenhui Wang, Li Dong, Yaru Hao, Shaohan Huang, Shuming Ma, Qixiang Ye, and Furu Wei. Grounding multimodal large language models to the world. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=lLmqxkfSIw.

Christopher Rawles, Alice Li, Daniel Rodriguez, Oriana Riva, and Timothy Lillicrap. Android in the wild: A large-scale dataset for android device control. *arXiv preprint arXiv:2307.10088*, 2023.

Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael S. Bernstein, Alexander C. Berg, and Li Fei-Fei. Imagenet large scale visual recognition challenge. *International Journal of Computer Vision*, 115: 211 – 252, 2014. URL https://api.semanticscholar.org/CorpusID:2930547.

Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade W Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, Patrick Schramowski, Srivatsa R Kundurthy, Katherine Crowson, Ludwig Schmidt, Robert Kaczmarczyk, and Jenia Jitsev. LAION-5b: An open large-scale dataset for training next generation image-text models. In *Thirty-sixth Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2022. URL https://openreview.net/forum?id=M3Y74vmsMcY.

Amanpreet Singh, Vivek Natarajan, Meet Shah, Yu Jiang, Xinlei Chen, Dhruv Batra, Devi Parikh, and Marcus Rohrbach. Towards vqa models that can read. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

Quan Sun, Qiying Yu, Yufeng Cui, Fan Zhang, Xiaosong Zhang, Yueze Wang, Hongcheng Gao, Jingjing Liu, Tiejun Huang, and Xinlong Wang. Emu: Generative pretraining in multimodality, 2024. URL https://arxiv.org/abs/2307.05222.

Zineng Tang, Ziyi Yang, Guoxin Wang, Yuwei Fang, Yang Liu, Chenguang Zhu, Michael Zeng, Chao-Yue Zhang, and Mohit Bansal. Unifying vision, text, and layout for universal document processing. *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 19254–19264, 2022. URL https://api.semanticscholar.org/CorpusID:254275326.

Gemma Team, Thomas Mesnard, Cassidy Hardin, Robert Dadashi, Surya Bhupatiraju, Shreya Pathak, Laurent Sifre, Morgane Rivière, Mihir Sanjay Kale, Juliette Love, Pouya Tafti, Léonard Hussenot, Pier Giuseppe Sessa, Aakanksha Chowdhery, Adam Roberts, Aditya Barua, Alex Botev, Alex Castro-Ros, Ambrose Slone, Amélie Héliou, Andrea Tacchetti, Anna Bulanova, Antonia Paterson, Beth Tsai, Bobak Shahriari, Charline Le Lan, Christopher A. Choquette-Choo, Clément Crepy, Daniel Cer, Daphne Ippolito, David Reid, Elena Buchatskaya, Eric Ni, Eric Noland, Geng Yan, George Tucker, George-Christian Muraru, Grigory Rozhdestvenskiy, Henryk Michalewski, Ian Tenney, Ivan Grishchenko, Jacob Austin, James Keeling, Jane Labanowski, Jean-Baptiste Lespiau, Jeff Stanway, Jenny Brennan, Jeremy Chen, Johan Ferret, Justin Chiu, Justin Mao-Jones, Katherine Lee, Kathy Yu, Katie Millican, Lars Lowe Sjoesund, Lisa Lee, Lucas Dixon, Machel Reid, Maciej

Mikuła, Mateo Wirth, Michael Sharman, Nikolai Chinaev, Nithum Thain, Olivier Bachem, Oscar Chang, Oscar Wahltinez, Paige Bailey, Paul Michel, Petko Yotov, Rahma Chaabouni, Ramona Comanescu, Reena Jana, Rohan Anil, Ross McIlroy, Ruibo Liu, Ryan Mullins, Samuel L Smith, Sebastian Borgeaud, Sertan Girgin, Sholto Douglas, Shree Pandya, Siamak Shakeri, Soham De, Ted Klimenko, Tom Hennigan, Vlad Feinberg, Wojciech Stokowiec, Yu hui Chen, Zafarali Ahmed, Zhitao Gong, Tris Warkentin, Ludovic Peran, Minh Giang, Clément Farabet, Oriol Vinyals, Jeff Dean, Koray Kavukcuoglu, Demis Hassabis, Zoubin Ghahramani, Douglas Eck, Joelle Barral, Fernando Pereira, Eli Collins, Armand Joulin, Noah Fiedel, Evan Senter, Alek Andreev, and Kathleen Kenealy. Gemma: Open models based on gemini research and technology, 2024. URL https://arxiv.org/abs/2403.08295.

OpenAI Team. Gpt-4 technical report, 2024.

Shengbang Tong, Ellis Brown, Penghao Wu, Sanghyun Woo, Manoj Middepogu, Sai Charitha Akula, Jihan Yang, Shusheng Yang, Adithya Iyer, Xichen Pan, Austin Wang, Rob Fergus, Yann LeCun, and Saining Xie. Cambrian-1: A fully open, vision-centric exploration of multimodal llms, 2024. URL https://arxiv.org/abs/2406.16860.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. Llama: Open and efficient foundation language models, 2023. URL https://arxiv.org/abs/2302.13971.

Bryan Wang, Gang Li, Xin Zhou, Zhourong Chen, Tovi Grossman, and Yang Li. Screen2words: Automatic mobile ui summarization with multimodal learning. *The 34th Annual ACM Symposium on User Interface Software and Technology*, 2021. URL https://api.semanticscholar.org/CorpusID:236957064.

Ke Wang, Junting Pan, Weikang Shi, Zimu Lu, Mingjie Zhan, and Hongsheng Li. Measuring multimodal mathematical reasoning with math-vision dataset, 2024a.

Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Yang Fan, Kai Dang, Mengfei Du, Xuancheng Ren, Rui Men, Dayiheng Liu, Chang Zhou, Jingren Zhou, and Junyang Lin. Qwen2-vl: Enhancing vision-language model's perception of the world at any resolution, 2024b. URL https://arxiv.org/abs/2409.12191.

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, 2024c.

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, 2022.

Yuan Yao, Tianyu Yu, Ao Zhang, Chongyi Wang, Junbo Cui, Hongji Zhu, Tianchi Cai, Haoyu Li, Weilin Zhao, Zhihui He, et al. Minicpm-v: A gpt-4v level mllm on your phone. *arXiv preprint arXiv:2408.01800*, 2024.

Jiabo Ye, Anwen Hu, Haiyang Xu, Qinghao Ye, Ming Yan, Yuhao Dan, Chenlin Zhao, Guohai Xu, Chenliang Li, Junfeng Tian, Qian Qi, Ji Zhang, and Fei Huang. mplug-docowl: Modularized multimodal large language model for document understanding, 2023a.

Jiabo Ye, Anwen Hu, Haiyang Xu, Qinghao Ye, Ming Yan, Guohai Xu, Chenliang Li, Junfeng Tian, Qi Qian, Ji Zhang, Qin Jin, Liang He, Xin Lin, and Fei Huang. UReader: Universal OCR-free visually-situated language understanding with multimodal large language model. In Houda Bouamor, Juan Pino, and Kalika Bali (eds.), *Findings of the Association for Computational Linguistics: EMNLP 2023*, pp. 2841–2858, Singapore, December 2023b. Association for Computational Linguistics. doi: 10.18653/v1/2023.findings-emnlp.187. URL https://aclanthology.org/2023.findings-emnlp.187.

Shukang Yin, Chaoyou Fu, Sirui Zhao, Ke Li, Xing Sun, Tong Xu, and Enhong Chen. A survey on multimodal large language models. *arXiv preprint arXiv:2306.13549*, 2023.

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. In *The Twelfth International Conference on Learning Representations*, 2024a. URL https://openreview.net/forum?id=2msbbX3ydD.

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 preprint arXiv:2404.05719*, 2024b.

Licheng Yu, Patrick Poirson, Shan Yang, Alexander C. Berg, and Tamara L. Berg. Modeling context in referring expressions, 2016. URL https://arxiv.org/abs/1608.00272.

Renrui Zhang, Jiaming Han, Chris Liu, Aojun Zhou, Pan Lu, Yu Qiao, Hongsheng Li, and Peng Gao. LLaMA-adapter: Efficient fine-tuning of large language models with zero-initialized attention. In *The Twelfth International Conference on Learning Representations*, 2024a. URL https://openreview.net/forum?id=d4UiXAHN2W.

Yi-Fan Zhang, Qingsong Wen, Chaoyou Fu, Xue Wang, Zhang Zhang, Liang Wang, and Rong Jin. Beyond llava-hd: Diving into high-resolution large multimodal models, 2024b. URL https://arxiv.org/abs/2406.08487.

Bo Zhao, Boya Wu, Muyang He, and Tiejun Huang. Svit: Scaling up visual instruction tuning, 2023. URL https://arxiv.org/abs/2307.04087.

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. In *The Twelfth International Conference on Learning Representations*, 2023.

Deyao Zhu, Jun Chen, Xiaoqian Shen, Xiang Li, and Mohamed Elhoseiny. MiniGPT-4: Enhancing vision-language understanding with advanced large language models. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=1tZbq88f27.

Table A: The training hyper-parameters used for fine-tuning UI-Pro in the experiments.

Hyper-Parameter	Value
Epoch	1
Global batch size	128
#GPUs	8
Learning rate for all stages	3e-5
weight decay	0.0
ADAM Beta2	0.95
Warm-up ratio	0.03
LR scheduler	Cosine
Model max length	2048
Frozen module	ViT
DeepSpeed	ZeRO-2
Data type	BFloat16

A Appendix

A Implementation Details

A.1 Data Curation Details

We found that certain datasets, such as RefCOCO and SeeClick include samples containing multi-turn dialogues while others do not, leading to potential imbalance issues caused by this multi-turn trait and resulting in unfair comparison. Additionally, excessively long dialogs exceed the context window of 2048 of UI-pro, causing training bugs. To resolve these issues, we reorganize the multi-turn dialogs to ensure each dialog contains no more than 650 text tokens to balance all samples.

A.2 Fine-tuning Details

The hyper-parameters of training UI-Pro is shown in Tab. A. All experiments are conducted with 8 L20 GPUs each with 48GB memory. The three-stage fine-tuning (stage 1: warming-up; stage 2: simple UI grounding task fine-tuning; stage 3: complex UI grounding task fine-tuning) spend approximately 22 hours in total.

Although the parameter numbers of the compressors can not be flexibly adjusted due to discrete parameter space and hardware efficiency issues, we try our best to match their sizes: (a) Merger: 10,488,832, (b) Resampler: 8,998,912, (c) C-Abstractor: 9,100,032, (d) H-Reducer: 10,489,344, (e) MLP: 8,392,704.

B Limitations

Despite the impressive UI element grounding capability, UI-Pro still encounters several limitations:

1. Model Diversity. This paper is targeted at LLaVA-like architectures that typically comprise a vision encoder, a vision-language connector, and an LLM. In practice, there exist various VLM architectures, such as Flamigo (Alayrac et al., 2022) and Emu (Sun et al., 2024) with multi-modal inputs and outputs. Future work can extend our findings to these architectures to generalize the insights.

2. UI Data Diversity. As UI-related datasets are much more scarce than natural image datasets, this work mainly conducts experiments with SeeClick and AutoGUi training datasets, which are the largest ones to date. This data insufficiency issue may be the cause of slight over-fitting observed in the data size scaling experiments, as shown in Fig 4. We hope future work can collect more diverse data to consolidate our findings.

3. Resource Intensiveness. Fine-tuning VLMs like UI-Pro can be extremely resource-intensive, requiring substantial computational power and time, which may limit accessibility for some researchers or developers. Due to such a resource restriction, this work uses small LLMs, i.e. Gemma-2B. We will extend our work to a larger scale by integrating larger LLMs like Llama-3-8B.