

UI-OCEANUS: Scaling GUI Agents with Synthetic Environmental Dynamics

Mengzhou Wu^{1,2}, Yuzhe Guo^{1,2}, Yuan Cao^{1,2}, Haochuan Lu³, Songhe Zhu³
Pingzhe Qu³, Xin Chen³, Kang Qin³, Zhongpu Wang³, Xiaode Zhang³
Xinyi Wang³, Wei Dai³, Gang Cao³, Yuetang Deng³, Zhi Gong³
Dezhi Ran^{†,*}, Linyi Li⁴, Wei Yang⁵, and Tao Xie^{*,1,2,5}

¹School of Computer Science, Peking University, Beijing, China

²Beijing Tongming Lake Information Technology Application Innovation Center, Beijing, China

³WeChat, Tencent Inc., China ⁴Simon Fraser University, Canada

⁵Institute of Systems for Advanced Computing, Fudan University, China

[†]Project Leader *Corresponding Authors

Abstract

Scaling generalist GUI agents is hindered by the high cost of human demonstrations and reliance on ungrounded synthetic teacher supervision. To transcend these limitations, we propose **UI-OCEANUS**, a framework that shifts the learning focus from mimicking high-level trajectories to learning interaction dynamics from executed transitions. Through a systematic investigation of self-supervised objectives, we identify that forward dynamics, defined as the generative prediction of future interface states, acts as the primary driver for scalability and outperforms inverse and backward objectives. **UI-OCEANUS** leverages this insight by converting low-cost autonomous exploration into high-density generative supervision to learn a semantic transition world model from executed transitions. Experimental evaluations across a series of models demonstrate consistent gains from our framework: models using Continual Pre-Training (CPT) on synthetic dynamics outperform non-CPT baselines by 7% on average in EM on the offline benchmark, increasing to an average relative improvement of 16.8% in success rate during real-world online navigation. Furthermore, we observe that navigation performance scales with synthetic data volume. These results suggest that grounding agents in forward predictive modeling offers a promising pathway to scalable GUI automation with cross-domain adaptability and compositional generalization.

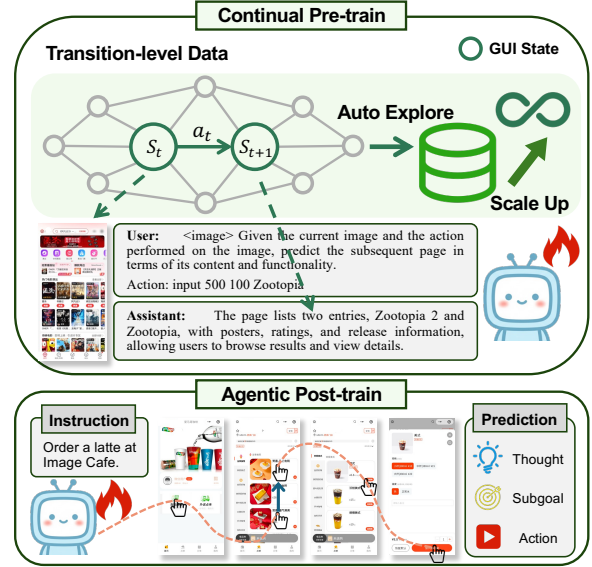


Figure 1: **Generalist GUI Agents via Scalable Transition World Model Learning.** (Top) We first establish a transition-aware representation by learning a forward dynamics objective. (Bottom) We then leverage this learned GUI transition world model to instantiate a generalist GUI agent through agentic post-training.

1 Introduction

Generalist GUI agents (Qin et al., 2025; Wang et al., 2025a; Wu et al., 2024b; Xu et al., 2025b; Hong et al., 2024; Ran et al., 2025a) traditionally rely on Behavioral Cloning (BC) (Torabi et al., 2018; Rawles et al., 2023; Zhang et al., 2024; Lu et al., 2025; Chen et al., 2025) to map visual observations to actions. However, acquiring trajectory-level data is prohibitively expensive. Unlike static image-text pairs, valid execution traces require long-horizon

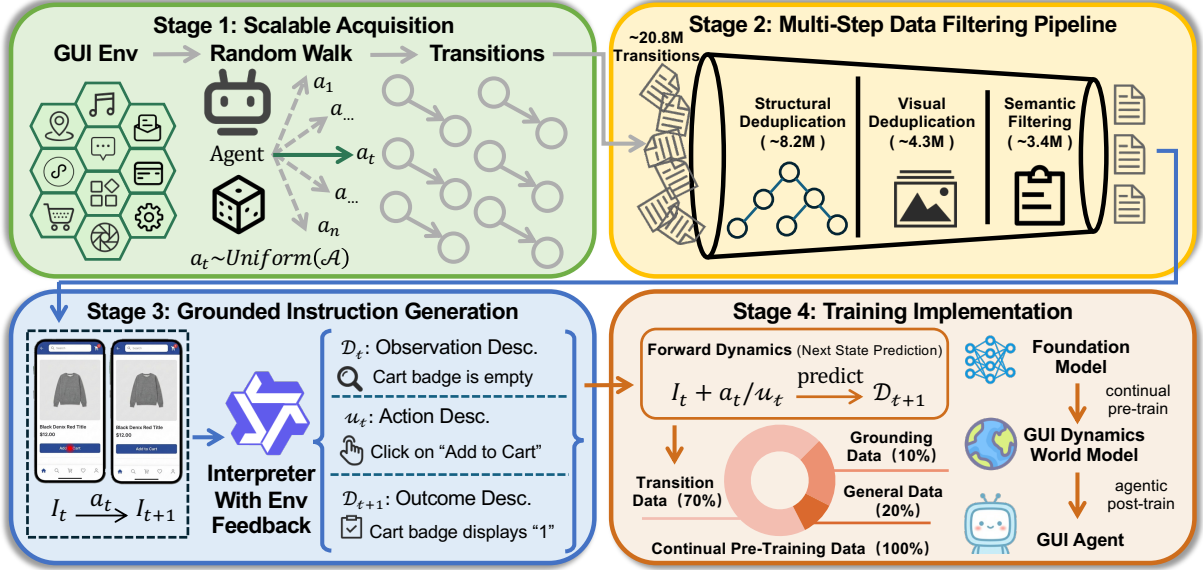


Figure 2: Overview of the Proposed UI-OCEANUS Framework. UI-OCEANUS consists of four sequential stages: (1) Scalable Acquisition; (2) Multi-Step Data Filtering Pipeline; (3) Grounded Instruction Generation; and (4) Training Implementation.

sequences where a single error can invalidate the entire trajectory. This sensitivity makes synthesizing reliable data difficult and necessitates labor-intensive human annotation (Zhang et al., 2025b; Rawles et al., 2023; Lu et al., 2025; Li et al., 2024), creating a major bottleneck for data scalability.

Existing methods (Tang et al., 2025; Wu et al., 2024b) seek to alleviate this bottleneck through distillation-based Continual Pre-Training (CPT) with static UI objectives such as captioning (Li et al., 2020) and grounding (Cheng et al., 2024), but such objectives provide limited supervision for learning interaction dynamics. Critically, these methods remain constrained by their reliance on ungrounded teacher supervision (Gou et al., 2021): they enhance an agent’s *static semantic understanding* but fail to capture the *temporal dynamics* of the environment. For instance, an agent might correctly identify a “submit” button (semantics) but stubbornly assume that clicking it leads to submission, failing to anticipate that an incomplete form will actually trigger a validation error (dynamics). Lacking such grounded interaction dynamics, these agents remain bound by static semantic priors, unable to adapt when dynamic environmental states contradict their shallow heuristics. Therefore, there is a pressing need for a scalable CPT objective that derives supervision directly from autonomous environmental feedback.

To address these limitations, we propose UI-OCEANUS, a self-supervised training framework

designed to equip agents with a GUI transition world model (Figure 1). The core insight in UI-OCEANUS is that the underlying set of atomic transitions (i.e., single-step observation-action-outcome tuples) in the GUI environment is both relatively *finite* and inherently *self-supervised*, providing a natural task-agnostic scaling dimension for CPT. The *finite* nature ensures that the environmental dynamics can be efficiently explored without an expert policy model or explicit task goal to reach high coverage. The *self-supervised* nature enables the environment to act as an execution-based verifier where the observed transition following the action serves as the observed outcome of an executed action. Both features enable agents to use task-agnostic exploration heuristics (Ran et al., 2024, 2023; Gu et al., 2019; Google, 2021; Su et al., 2017) to efficiently harvest vast quantities of these atomic transitions. Crucially, these atomic transitions ground the learning process in the *interaction dynamics* of the interface before post-training. This separation effectively decouples the acquisition of interaction dynamics from the execution of user intent: the agent learns “what is possible” (dynamics) from massive data before learning “what is desirable” (intent) from scarce expert demonstrations.

Building on this insight, we develop a scalable training pipeline that converts low-cost autonomous exploration into dense forward-dynamics supervision and incorporates this supervision into a two-stage model-training process. We investi-

gate the spectrum of environmental dynamics and find that forward dynamics, defined as the generative prediction of future interface states, serves as the key objective enabling performance scaling with data volume, outperforming inverse dynamics and backward dynamics. Consequently, **UI-OCEANUS** prioritizes *forward dynamics prediction* as the generative objective (Ha and Schmidhuber, 2018; Hafner et al., 2020). Technically, we capture raw interaction traces, isolate unique and meaningful atomic transitions via strict hash-based filtering and semantic filtering, and use a Vision-Language Model (VLM) to synthesize precise descriptions of environmental changes. Finally, we implement a two-stage training strategy: the model is first continually pre-trained on a mixture of GUI dynamics, grounding, and general data to learn a semantic transition world model, and then specialized via agentic post-training to align this transition-aware prior with complex instruction following.

We conduct extensive empirical evaluations within the WeChat mini-program ecosystem (Tencent, 2024), one of the world’s largest and most diverse digital platforms. Hosting millions of distinct mini-programs for over a billion monthly active users (MAUs), this ecosystem presents a virtually infinite space of interaction logic and visual layouts, demanding that agents capture transferable UI transition patterns rather than memorize specific application interfaces. Notably, **UI-OCEANUS** demonstrates consistent scalability on the WeChat mini-program offline benchmark, achieving an average relative EM improvement of 7% across seven different VLM backbones compared with strong baselines using identical Supervised Fine-Tuning. Crucially, these benefits are amplified in online navigation: **UI-OCEANUS** boosts the real-world success rate by 21.9% during the cold-start phase and maintains an average 15% lead even after both step-level and multi-step GRPO (Shao et al., 2024) alignment. Furthermore, our experiments yield three critical insights: (1) **Data scaling**: Across all seven Qwen-VL models, performance is the highest at the 100% CPT scale, showing an overall positive data-scaling trend with no observed saturation within the evaluated data range. (2) **Mechanism**: Forward dynamics is more effective than inverse dynamics, as its higher predictive difficulty encourages the model to learn stronger semantic visual representations. (3) **Generalization**: The learned GUI transition world model captures transferable UI transition patterns effectively, enabling general-

ization to unseen mini-programs and Android environments, as well as compositional reasoning on complex multi-step tasks never seen during training.

Our main contributions are as follows:

- We pioneer a shift in GUI agent learning by separating the acquisition of atomic interaction dynamics from high-level instruction following. To the best of our knowledge, our work is the first to systematically scale environment dynamics as a self-supervised CPT objective for GUI agents, bypassing the data scalability limitations of trajectory-based behavioral cloning.
- We propose **UI-OCEANUS**, a scalable framework designed to mine training supervision directly from observed environmental transitions.
- We conduct extensive evaluations validating the performance scaling of **UI-OCEANUS**. Our results demonstrate generalization across both domains and horizons.

2 Background and Related Work

Generalist GUI agents autonomously execute user instructions by interacting with GUIs, thereby improving user productivity in daily tasks (Wang et al., 2024). Built on pretrained VLMs, generalist GUI agents are post-trained using Behavioral Cloning (BC) (Wu et al., 2024b; Lin et al., 2025a; Lu et al., 2025; Chen et al., 2025) or multi-step Reinforcement Learning (RL) (Ye et al., 2025; Xu et al., 2025a; Wang et al., 2025a). However, post-training methods struggle under the capability ceiling of the base model (Zhang et al., 2025a; Yue et al., 2025), underscoring the necessity of Continual Pre-Training (CPT) prior to policy alignment. A fundamental challenge for CPT is the data scalability bottleneck: collecting high-quality human demonstrations is prohibitively expensive. To bypass this challenge, existing research mainly uses two types of CPT tasks: (1) static UI-understanding tasks, such as element grounding or screen captioning (Tang et al., 2025; Wu et al., 2024b, 2025), which require powerful foundation models to label GUI screenshots and lack grounding in real execution; and (2) dynamic interaction with environments under instructions (Lin et al., 2025b; Sun et al., 2025; Ramrakhya et al., 2025; Pahuja et al., 2025; Cao et al., 2026), which notoriously

yields few high-quality long-horizon samples. **UI-OCEANUS** introduces a new CPT task: learning the forward dynamics of atomic transitions. This dynamic learning task enables **UI-OCEANUS** to reduce reliance on ungrounded teacher supervision while avoiding the fragility of long-horizon trajectory synthesis.

Conceptualizing the GUI as a dynamic system aligns with the broader literature on world models. In the specific GUI domain, prior work has explored this direction through visual state prediction (Luo et al., 2025; Wang et al., 2025b) and action prediction from state changes (Gao et al., 2026). Unlike Early Experience (Zhang et al., 2025c), which branches from expert-trajectory states, **UI-OCEANUS** autonomously collects task-agnostic transitions without expert demonstrations or task goals and systematically compares three dynamics objectives. However, the field still lacks a scalable, task-agnostic CPT objective for learning broad transition patterns. We distinguish our work by leveraging autonomous exploration and intrinsic environmental feedback to learn a GUI transition world model via CPT, establishing a transition-aware representation before agentic post-training.

See Appendix B for further discussion.

3 The UI-OCEANUS Framework

To instantiate the transition formulation defined in Appendix A, we introduce **UI-OCEANUS**, a unified framework for learning GUI dynamics from large-scale autonomous interaction. As illustrated in Figure 2, **UI-OCEANUS** consists of the following four sequential stages.

3.1 Scalable Acquisition via Autonomous Exploration

We autonomously explore the WeChat mini-program ecosystem to harvest open-domain transitions. Using a distributed fleet of 50 nodes, we use a structured random walk policy: agents interact with UI elements identified via accessibility trees without human intervention. For each step t , we record the transition tuple (s_t, a_t, s_{t+1}) , where $s_t = (I_t, X_t)$ denotes the screenshot and structural metadata. This process yields over 20.8M raw transitions over one month.

3.2 Multi-Step Data Filtering Pipeline

We refine $\mathcal{D}_{\text{raw}}$ into a high-quality corpus $\mathcal{D}_{\text{cpt}}$ via a three-stage pipeline, with implementation details

and filtering analysis provided in Appendix D. (1) **Structural Deduplication:** MinHash signatures on accessibility trees remove repeated content templates, reducing data to 8.2M. (2) **Visual Deduplication:** Pixel-level pHash/dHash methods identify visually static transitions, narrowing the corpus to 4.3M. (3) **Semantic Filtering:** A VLM filters system errors and rendering artifacts by verifying action-feedback consistency, resulting in a final set of 3.4M transitions.

3.3 Grounded Instruction Generation

For each filtered transition, we use a VLM interpreter (Qwen3-VL-235B-A22B-Instruct (Bai et al., 2025a)) to verbalize the observed pre-state, executed action, and post-state. Importantly, the interpreter does not synthesize the transition itself: both screenshots come from actual system execution. The resulting supervision trains the model to predict a textual description of the next UI state given the current screen and action. Appendix G analyzes the sensitivity of this interpreter stage to teacher model scale.

By generating instructions directly grounded in actual environmental state transitions, this stage produces large-scale, reliable, and high-fidelity data for CPT.

3.4 Training Implementation

3.4.1 Continual Pre-Training

Using the curated dataset $\mathcal{D}_{\text{cpt}}$ and generated grounded instructions, we construct a unified training corpus designed to facilitate the model’s understanding of GUI dynamics.

We define three distinct categories of training tasks to comprehensively capture GUI dynamics:

Forward Dynamics: Given the screenshot of an initial state and an action (either as an exact action or a natural-language summary), the model predicts the resulting GUI state description. *Input:* $\mathcal{V} = [I_t, a_t]$ or $\mathcal{V} = [I_t, u_t]$; *Target:* $\mathbf{y} = \mathcal{D}_{t+1}$.

Inverse Dynamics: Given screenshots of the initial and resulting states, the model predicts the action connecting them. *Input:* $\mathcal{V} = [I_t, I_{t+1}]$; *Target:* $\mathbf{y} = u_t$ or $\mathbf{y} = a_t$.

Backward Dynamics: Given the screenshot of a resulting state and an action summary, the model predicts a description of the initial GUI state. *Input:* $\mathcal{V} = [u_t, I_{t+1}]$; *Target:* $\mathbf{y} = \mathcal{D}_t$.

Data Mixing. We construct the final training set $\mathcal{D}_{\text{mix}}$ by strategically combining three distinct data sources: (1) GUI dynamics (70%): Informed

Table 1: **Data and Model Scaling Results of UI-OCEANUS.** Evaluation of Exact Match (EM) scores on the held-out offline benchmark. **UI-OCEANUS** achieves consistent performance gains across all seven Qwen-VL models.

Model	Size	Reference		UI-OCEANUS Data Scale (+ SFT)				
		Base (Zero-shot)	w/ grounding	0% (SFT)	12.5%	25%	50%	100% (Ours)
<i>I. Qwen3-VL Series with UI-OCEANUS</i>								
Qwen3-VL	2B	2.4	36.5	47.9	49.4	51.3	51.7	52.7
Qwen3-VL	4B	41.7	52.0	57.3	58.3	59.0	59.3	60.0
Qwen3-VL	8B	42.3	54.9	58.1	58.0	59.6	60.4	60.8
Qwen3-VL	32B	47.1	61.2	61.5	61.0	63.7	62.5	65.0
<i>II. Qwen2.5-VL Series with UI-OCEANUS</i>								
Qwen2.5-VL	3B	1.1	27.9	47.7	49.2	48.5	49.6	50.6
Qwen2.5-VL	7B	13.0	37.8	51.0	53.5	53.3	53.4	55.7
Qwen2.5-VL	32B	33.9	56.5	60.5	61.6	63.2	64.2	64.8

by our ablation study (Section 4.3), we exclusively use forward dynamics ($I_t, a_t \rightarrow \mathcal{D}_{t+1}$) as the semantic transition-modeling objective; (2) General multimodal data (20%), which acts as a regularizer against catastrophic forgetting during GUI-specialized CPT; and (3) UI grounding data (10%). Appendix I reports evaluations on 11 public multimodal and reasoning benchmarks, showing largely preserved general capabilities with mild trade-offs on several tasks. We detail the specific training hyperparameters in Appendix H.1.

3.4.2 Agentic Post-Training

Following the CPT phase, we conduct an agentic post-training stage. In this phase, the model is fine-tuned on high-quality GUI agent trajectories to explicitly align the GUI transition world model with user intent, bridging the gap between transition-aware representations and task execution. We detail the specific training hyperparameters in Appendix H.2.

4 Experiments

In this section, we systematically evaluate the **UI-OCEANUS** framework. Our evaluation is structured to answer four critical questions: (1) **Scalability.** Can autonomous exploration serve as a scalable data source that yields consistent performance gains? (Section 4.1) (2) **Online efficacy.** Does the learned GUI transition world model translate effectively to real-world online navigation and act as a persistent prior that synergizes with agen-

tic post-training? (Section 4.2) (3) **Mechanism.** Which self-supervised objective yields the most significant performance gains for downstream GUI navigation? (Section 4.3) (4) **Transfer and generalization.** Does the learned GUI transition world model transfer beyond the WeChat ecosystem and support compositional transition prediction? (Section 4.4)

4.1 Data and Model Scaling of UI-OCEANUS

A central premise of our framework is that autonomous environmental feedback can overcome the data scarcity bottleneck. We first investigate the scaling properties of **UI-OCEANUS**.

4.1.1 Setup

Training Configurations. During the CPT phase, we use the data mixture detailed in Section 3.4, totaling approximately 5M samples (3.2B tokens). For the subsequent agentic post-training phase, we use 8K high-quality GUI navigation samples. To study data scaling, we vary the volume of CPT data injected from 0% to 100%.

Evaluation Protocol. To support large-scale experiments, we evaluate performance on a held-out offline benchmark comprising 8K diverse mini-program tasks not seen during training, complementing prior public and industrial mobile UI navigation evaluations (Rawles et al., 2025; Ran et al., 2025b). Detailed dataset statistics are provided in Appendix J.1. Following prior work (Wu et al., 2024b; Zhang et al., 2024; Li et al., 2024), we report Exact Match (EM), which requires both the

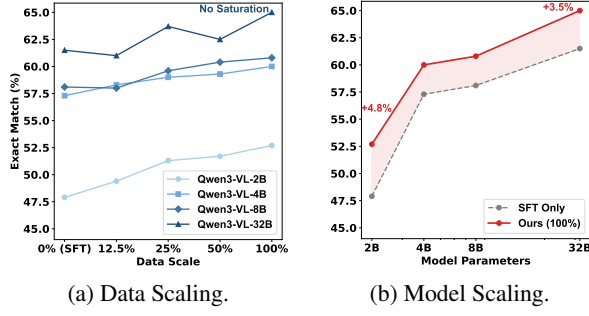


Figure 3: Data and Model Scaling Behavior of **UI-OCEANUS** on the Qwen3-VL Series.

action type and its parameters to be correct. We assess performance across Qwen3-VL (Bai et al., 2025a) (2B, 4B, 8B, 32B) and Qwen2.5-VL (Bai et al., 2025b) (3B, 7B, 32B) families. We compare our method against two distinct baselines. (1) **Base Models:** We evaluate raw base models in a zero-shot setting. To decouple planning from grounding, we also report a “w/ grounding” setting, where an external specialized grounding model based on Qwen2.5-VL-32B maps natural-language action descriptions to executable actions; (2) **SFT-only Baselines:** Models fine-tuned solely on the post-training data without any dynamics CPT (denoted as 0% data scale).

4.1.2 Results

As presented in Table 1 and visualized in Figure 3, our empirical findings validate scaling trends for GUI transition world models across both data and model dimensions.

Data Scaling. We observe a consistent positive correlation between the volume of CPT dynamics data and downstream performance. Starting from the standard SFT baseline (0% data scale), adding our dynamics data yields consistent improvements across all model sizes, confirming that autonomous exploration serves as a scalable data source.

Model Scaling. **UI-OCEANUS** demonstrates consistent scalability with model size. The performance gains over the SFT baseline remain consistent across all scales. This result indicates that the learned interaction dynamics provide transferable transition-aware representations whose benefits do not diminish as model size grows.

Scaling at the Largest Evaluated Data Scale.

Across all seven Qwen-VL models, performance is the highest at the 100% CPT scale. These results show an overall positive data-scaling trend, with no observed saturation within the evaluated data range.

Table 2: Online Evaluation across Training Stages.

Training Stage	SR (%)		Δ
	w/o CPT	UI-OCEANUS	
<i>Qwen3-VL-8B</i>			
SFT + Step-level GRPO	27.5	30.9	+12.4%
<i>Qwen2.5-VL-32B</i>			
SFT (Cold Start)	24.2	29.5	+21.9%
+ Step-level GRPO	32.2	38.9	+20.8%
+ Multi-step GRPO	39.6	44.3	+11.9%

This pattern suggests that further scaling may yield additional gains.

4.2 Online Evaluation: Benefits Across Post-Training Stages

While offline metrics quantify predictive accuracy, the ultimate test lies in online execution. We investigate the effectiveness of **UI-OCEANUS** when integrated into a progressive post-training pipeline.

4.2.1 Setup

Training Configurations. We compare agents initialized with **UI-OCEANUS** CPT against agents initialized without CPT. The post-training pipeline consists of three progressive stages: (1) **SFT (cold start):** Supervised fine-tuning using simple samples from the SFT dataset to establish basic instruction-following capabilities; (2) **Step-level GRPO:** Optimization on hard samples from the SFT dataset using dense step-wise rewards to refine complex reasoning; (3) **Multi-step GRPO:** Further alignment using approximately 1K instructions in the real environment and a VLM outcome reward model that provides sparse trajectory-level rewards.

Evaluation Protocol. We report the success rate (SR) on a held-out benchmark of 149 instructions covering diverse mini-programs (refer to Appendix J.2 for a detailed breakdown of domain coverage and task complexity). All evaluation episodes are executed in the real online environment, and task success is manually verified by human annotators to ensure rigorous correctness.

4.2.2 Results

As shown in Table 2, **UI-OCEANUS** provides a strong transition-aware initialization that benefits the entire post-training pipeline.

Consistent Gains across Post-Training Stages.

UI-OCEANUS establishes an immediate advantage in the initial SFT phase and, crucially, maintains this lead throughout the subsequent RL align-

Table 3: **Ablation Study on Continual Pre-Training Objectives.** We compare different dynamics tasks against the non-CPT baseline. Forward dynamics consistently achieves the highest performance across all scales.

Dynamics	Formulation	Qwen3-VL-2B		Qwen3-VL-4B		Qwen2.5-VL-3B		Overall	
		TM	EM	TM	EM	TM	EM	TM	EM
Baseline (w/o CPT)		75.9	47.9	79.2	57.3	75.7	47.7	76.9	51.0
Forward	$I_t, u_t \rightarrow \mathcal{D}_{t+1}$	77.7	52.7	81.0	60.3	<u>76.7</u>	48.8	78.5	<u>53.9</u>
	$I_t, a_t \rightarrow \mathcal{D}_{t+1}$	<u>77.6</u>	52.7	<u>80.3</u>	<u>60.0</u>	76.9	50.6	<u>78.3</u>	54.4
Inverse	$I_t, I_{t+1} \rightarrow u_t$	75.8	49.5	75.8	49.5	75.3	<u>49.0</u>	75.6	49.3
	$I_t, I_{t+1} \rightarrow a_t$	75.6	46.4	77.8	50.9	75.6	47.2	76.3	48.2
	$I_t, \mathcal{D}_{t+1} \rightarrow u_t$	76.3	50.2	79.0	57.4	76.2	48.3	77.2	52.0
	$I_t, \mathcal{D}_{t+1} \rightarrow a_t$	75.7	46.6	78.0	52.5	75.8	47.6	76.5	48.9
Backward	$u_t, I_{t+1} \rightarrow \mathcal{D}_t$	77.1	<u>50.4</u>	80.0	58.7	76.3	47.5	77.8	52.2

ment steps. This persistence demonstrates that the transition-aware prior is not overwritten by optimization but rather facilitates more efficient policy learning.

Amplified Benefits in Online Environments. Notably, the performance gains yielded by **UI-OCEANUS** are more pronounced in the online setting than on the offline benchmark (Section 4.1), highlighting that our GUI transition world model equips an agent with a transition-aware prior useful for real-world interaction. With only 32B parameters, our best model achieves a 44.3% online success rate, outperforming Seed-1.8 by 16.8 percentage points and approaching the leading proprietary models Claude-4.5-Opus at 46.3% and Gemini-3-Flash at 49.0%, with gaps of only 2.0 and 4.7 percentage points, respectively (Table 17).

4.3 Mechanism: Forward Dynamics Helps the Most

We dissect the contributions of different CPT objectives to understand the source of the performance gains.

4.3.1 Setup

Training Configuration. To isolate the impact of objective formulation, we conduct controlled experiments on three models (Qwen3-VL-2B, Qwen3-VL-4B, and Qwen2.5-VL-3B). We compare the three distinct categories of self-supervised tasks detailed in Section 3.4, while keeping the underlying corpus of UI transitions constant (5M samples).

Evaluation Protocol. We use the same SFT and evaluation settings as in Section 4.1. In addition to the previously defined EM, we report Type Match (TM), which evaluates only whether the predicted action type matches the ground truth.

4.3.2 Results and Analysis

As detailed in Table 3, our ablation study reveals critical insights into how different transition objectives shape an agent’s representations.

The Effectiveness of Forward Dynamics. The simplest forward-dynamics formulation, $I_t, a_t \rightarrow \mathcal{D}_{t+1}$, achieves the highest EM (54.4%), outperforming both standard inverse dynamics (48.2%) and backward dynamics (52.2%). Predicting “*what happens next*” forces the model to encode the state transition logic of the environment. Unlike methods using separate world models for planning (Luo et al., 2025; Li et al., 2025), our method uses this transition objective to shape an agent’s representations directly, allowing the agent to benefit from transition-aware features without adding an external planning model.

Why Inverse Dynamics Lags. The weaker performance of standard inverse dynamics is partly tied to its simpler target and different output-token distribution: it predicts sparse action labels and coordinates, while forward dynamics predicts richer future-state descriptions. The training loss curves in Figure 4 (Appendix E) are consistent with this distinction, showing faster convergence for inverse dynamics under sparse action targets. We therefore treat the gap as an objective-level comparison rather than evidence that temporal direction alone explains performance. Additional target-density controls are reported in Appendix F; they preserve the advantage of future-state prediction.

4.4 Cross-Platform Transfer and Compositional Generalization

We evaluate two complementary forms of generalization: controlled transition generalization over

one- and two-step UI transitions, followed by transfer to the public AndroidWorld benchmark.

4.4.1 Controlled Transition Generalization Setup

We further investigate whether an agent has merely memorized specific transition pairs or learned transition patterns capable of cross-domain transfer and compositional reasoning. To distinguish between these possibilities, we define two levels of tasks: Level 1 (L1, atomic) uses standard single-step transitions ($t \rightarrow t+1$) to evaluate cross-domain transfer performance; Level 2 (L2, compositional) introduces multi-step chaining ($t \rightarrow t+2$) that is never seen during training to assess compositional generalization, testing the ability to combine atomic GUI transition patterns for long-horizon prediction. We construct a controlled evaluation benchmark using atomic transitions from mini-programs, partitioned at the application level into seen and unseen groups. We randomly sample 1.2M transitions for training, while reserving 10K samples for L1 test sets and constructing 10K samples for L2 test sets within each group. To evaluate out-of-distribution (OOD) robustness, we additionally sample 10K random transitions from AndroidControl.

Training Configurations. We train two specialized variants of **UI-OCEANUS-7B** using the same data mixture as in Section 4.3. Specifically, **UI-OCEANUS-7B-F** is trained on L1 forward tasks ($I_t, u_t \rightarrow \mathcal{D}_{t+1}$), and **UI-OCEANUS-7B-I** is trained on L1 inverse tasks ($I_t, I_{t+1} \rightarrow u_t$). Both models are trained on approximately 1.7M total samples.

Evaluation Protocol. We use a VLM-as-a-judge protocol to handle the open-ended nature of generative predictions. For forward dynamics, we explicitly query the model to predict five distinct UI elements of the outcome page to facilitate objective verification via a VLM judge. For inverse dynamics, the model infers the first action given the start and end states, with correctness verified by a VLM judge.

4.4.2 Controlled Transition Generalization Results

As shown in Table 4, **UI-OCEANUS** performs consistently across both domain shifts and reasoning horizons.

Cross-Domain Generalization. Our model exhibits strong transfer capabilities across distribution shifts. Notably, on L1 Forward tasks,

Table 4: **Cross-Domain and Compositional Generalization.** We evaluate the learned GUI transition world model across varying distribution shifts and reasoning horizons.

Lvl	Task	Model	Seen mini-P.	Unseen mini-P.	OOD Android
L1	Forward	Qwen2.5-VL-7B	41.2	43.1	46.5
	I_t, u_t	Qwen2.5-VL-32B	48.5	49.6	65.8
	$\rightarrow \mathcal{D}_{t+1}$	UI-OCEANUS-7B-F	53.1	56.9	49.1
	Inverse	Qwen2.5-VL-7B	17.0	18.4	31.1
L2	I_t, I_{t+1}	Qwen2.5-VL-32B	37.0	37.5	67.3
	$\rightarrow u_t$	UI-OCEANUS-7B-I	62.3	58.9	55.3
	Forward	Qwen2.5-VL-7B	41.3	41.1	—
	I_t, u_t, u_{t+1}	Qwen2.5-VL-32B	46.9	46.3	—
L2	$\rightarrow \mathcal{D}_{t+2}$	UI-OCEANUS-7B-F	47.0	47.2	—
	Inverse	Qwen2.5-VL-7B	6.7	6.2	—
	I_t, I_{t+2}	Qwen2.5-VL-32B	26.5	26.1	—
	$\rightarrow u_t$	UI-OCEANUS-7B-I	38.7	37.6	—

Table 5: **AndroidWorld Transfer.** **UI-OCEANUS** uses only mini-program transitions; both rows use identical AndroidControl SFT data.

Model	Base+SFT	UI-OCEANUS	Δ
Qwen2.5-VL-7B	11.2	14.7	+31.3%
Qwen2.5-VL-32B	19.8	24.1	+21.7%

UI-OCEANUS-7B achieves 56.9 on unseen mini-programs, surpassing the larger Base-32B (49.6). This result indicates that our method learns transferable interaction patterns rather than memorizing application-specific layouts. Furthermore, on the challenging OOD Android transition benchmark, **UI-OCEANUS-7B** consistently outperforms its base model across all tasks, suggesting transfer beyond mini-program layouts in this controlled setting.

Compositional Generalization. Crucially, despite being trained strictly on single-step transitions, **UI-OCEANUS** exhibits strong compositional generalization on Level 2 tasks. This result implies that **UI-OCEANUS** does not merely memorize single-step transitions but enables an agent to mentally “chain” atomic GUI transition patterns. This capability supports multi-step consequence prediction needed for long-horizon planning.

4.4.3 Cross-Platform Transfer on AndroidWorld

To assess cross-platform transfer on a public benchmark, we evaluate matched Qwen2.5-VL baselines on AndroidWorld after identical SFT on AndroidControl. **UI-OCEANUS** uses the same SFT data but is initialized with CPT performed exclusively on WeChat mini-program transitions.

As shown in Table 5, despite using no Android transitions during CPT and only lightweight SFT on AndroidControl, **UI-OCEANUS** achieves consistent gains across both model scales, suggesting that the learned transition representations capture interaction patterns that transfer beyond the source ecosystem.

5 Conclusion

We have introduced **UI-OCEANUS**, a framework that overcomes the data scalability bottleneck by learning a GUI transition world model from autonomous exploration. By shifting from trajectory imitation to learning atomic interaction dynamics, we unlock a scalable CPT supervision source independent of human annotation. Our results identify forward dynamics as the critical objective for this capability and show an overall positive data-scaling trend across 7 Qwen-VL models and 35 model-data configurations, with no observed saturation within the evaluated data range. By separating the acquisition of interaction dynamics from the execution of user intent, **UI-OCEANUS** provides a scalable pathway toward generalist GUI agents.

6 Limitations

UI-OCEANUS predicts textual descriptions of future GUI states rather than executable simulator states, visual tokens, or DOM-level representations. While this semantic objective aligns naturally with language-conditioned GUI agents, it may omit fine-grained spatial and rendering dynamics. The CPT corpus and strongest empirical evidence come from the WeChat mini-program ecosystem, while broader validation on public GUI environments remains as future work.

Acknowledgments

This work was partially supported by the Fundamental and Interdisciplinary Disciplines Breakthrough Plan of the Ministry of Education of China (No. JYB2025XDXM118) and the National Natural Science Foundation of China under Grant Nos. 623B2006, U25A6023, and 92464301. Tao Xie is also affiliated with the Key Laboratory of High Confidence Software Technologies (Peking University), Ministry of Education, China; the Shanghai Institute of Systems for Open Computing, China; and the School of Computer Science and Engineering, Southeast University, China.

References

- Anthropic. 2025a. [Claude Opus 4.5](#). Accessed: 2026-01-29.
- Anthropic. 2025b. [Claude Sonnet 4.5](#). Accessed: 2026-01-29.
- Mahmoud Assran, Quentin Duval, Ishan Misra, Piotr Bojanowski, Pascal Vincent, Michael Rabbat, Yann LeCun, and Nicolas Ballas. 2023. Self-supervised learning from images with a joint-embedding predictive architecture. In *CVPR*, pages 15619–15629.
- Shuai Bai, Yuxuan Cai, Ruizhe Chen, Keqin Chen, Xionghui Chen, Zesen Cheng, Lianghao Deng, Wei Ding, Chang Gao, Chunjiang Ge, Wenbin Ge, Zhi-fang Guo, Qidong Huang, Jie Huang, Fei Huang, Binyuan Hui, Shutong Jiang, Zhaohai Li, Mingsheng Li, and 45 others. 2025a. Qwen3-VL technical report. *arXiv preprint arXiv:2511.21631*.
- Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibao Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, Humen Zhong, Yuanzhi Zhu, Mingkun Yang, Zhaohai Li, Jianqiang Wan, Pengfei Wang, Wei Ding, Zheren Fu, Yiheng Xu, and 8 others. 2025b. Qwen2.5-VL technical report. *arXiv preprint arXiv:2502.13923*.
- Tim Brooks, Aleksander Holynski, and Alexei A. Efros. 2023. InstructPix2Pix: Learning to follow image editing instructions. In *CVPR*, pages 18392–18402.
- ByteDance Seed. 2025. [Seed1.8 model card: Towards generalized real-world agency](#). Accessed: 2026-01-29.
- Yuan Cao, Dezhi Ran, Mengzhou Wu, Yuzhe Guo, Xin Chen, Ang Li, Gang Cao, Zhi Gong, Hao Yu, Linyi Li, Wei Yang, and Tao Xie. 2026. GUI-GENESIS: Automated synthesis of efficient environments with verifiable rewards for GUI agent post-training. *arXiv preprint arXiv:2602.14093*.
- Wentong Chen, Junbo Cui, Jinyi Hu, Yujia Qin, Junjie Fang, Yue Zhao, Chongyi Wang, Jun Liu, Guirong Chen, Yupeng Huo, Yuan Yao, Yankai Lin, Zhiyuan Liu, and Maosong Sun. 2025. [GUICourse: From general vision language model to versatile GUI agent](#). In *ACL*, pages 21936–21959.
- Kanzhi Cheng, Qiushi Sun, Yougang Chu, Fangzhi Xu, Yantao Li, Jianbing Zhang, and Zhiyong Wu. 2024. [SeeClick: Harnessing GUI grounding for advanced visual GUI agents](#). In *ACL*, pages 9313–9332.
- Longxi Gao, Li Zhang, Pengzhi Gao, Wei Liu, Jian Luan, and Mengwei Xu. 2026. GUI-Shift: Enhancing VLM-based GUI agents through self-supervised reinforcement learning. In *ICLR*, volume 2026, pages 39236–39255.
- Google. 2021. Android Monkey. <https://developer.android.com/studio/test/monkey>. Accessed: 2026-01-03.

- Google. 2025. [Gemini 3 Flash](#). Accessed: 2026-01-29.
- Jianping Gou, Baosheng Yu, Stephen J Maybank, and Dacheng Tao. 2021. Knowledge distillation: A survey. *International Journal of Computer Vision*, 129(6):1789–1819.
- Tianxiao Gu, Chengnian Sun, Xiaoxing Ma, Chun Cao, Chang Xu, Yuan Yao, Qirun Zhang, Jian Lu, and Zhendong Su. 2019. Practical GUI testing of Android applications via model abstraction and refinement. In *ICSE*, pages 269–280.
- David Ha and Jürgen Schmidhuber. 2018. World models. *arXiv preprint arXiv:1803.10122*, 2(3).
- Danijar Hafner, Timothy Lillicrap, Jimmy Ba, and Mohammad Norouzi. 2020. [Dream to control: Learning behaviors by latent imagination](#). In *ICLR*.
- Danijar Hafner, Jurgis Pasukonis, Jimmy Ba, and Timothy Lillicrap. 2025. [Mastering diverse control tasks through world models](#). *Nature*, 640:647–653.
- Wenyi Hong, Weihan Wang, Qingsong Lv, Jiazheng Xu, Wenmeng Yu, Junhui Ji, Yan Wang, Zihan Wang, Yuxiao Dong, Ming Ding, and Jie Tang. 2024. CoAgent: A visual language model for GUI agents. In *CVPR*, pages 14281–14290.
- Yuzhou Huang, Liangbin Xie, Xintao Wang, Ziyang Yuan, Xiaodong Cun, Yixiao Ge, Jiantao Zhou, Chao Dong, Rui Huang, Ruimao Zhang, and Ying Shan. 2024. SmartEdit: Exploring complex instruction-based image editing with multimodal large language models. In *CVPR*, pages 8362–8371.
- Yann LeCun. 2022. [A path towards autonomous machine intelligence](#). Version 0.9.2.
- Shufan Li, Konstantinos Kallidromitis, Akash Gokul, Yusuke Kato, Kazuki Kozuka, and Aditya Grover. 2025. MobileWorldBench: Towards semantic world modeling for mobile agents. *arXiv preprint arXiv:2512.14014*.
- Wei Li, William E Bishop, Alice Li, Christopher Rawles, Folawiyo Campbell-Ajala, Divya Tyamagundlu, and Oriana Riva. 2024. On the effects of data scale on UI control agents. *NeurIPS*, 37:92130–92154.
- Yang Li, Gang Li, Luheng He, Jingjie Zheng, Hong Li, and Zhiwei Guan. 2020. Widget captioning: Generating natural language description for mobile user interface elements. *arXiv preprint arXiv:2010.04295*.
- Kevin Qinghong Lin, Linjie Li, Difei Gao, Zhengyuan Yang, Shiwei Wu, Zechen Bai, Stan Weixian Lei, Lijuan Wang, and Mike Zheng Shou. 2025a. ShowUI: One vision-language-action model for GUI visual agent. In *CVPR*, pages 19498–19508.
- Musen Lin, Minghao Liu, Taoran Lu, Lichen Yuan, Yiwei Liu, Haonan Xu, Yu Miao, Yuhao Chao, and Zhaojian Li. 2025b. GUI-ReWalk: Massive data generation for GUI agent via stochastic exploration and intent-aware reasoning. *arXiv preprint arXiv:2509.15738*.
- Quanfeng Lu, Wenqi Shao, Zitao Liu, Lingxiao Du, Fanqing Meng, Boxuan Li, Botong Chen, Siyuan Huang, Kaipeng Zhang, and Ping Luo. 2025. [GUIOdyssey: A comprehensive dataset for cross-app GUI navigation on mobile devices](#). In *ICCV*, pages 22404–22414.
- Dezhao Luo, Bohan Tang, Kang Li, Georgios Papoudakis, Jifei Song, Shaogang Gong, Jianye Hao, Jun Wang, and Kun Shao. 2025. ViMo: A generative visual GUI world model for app agents. *arXiv preprint arXiv:2504.13936*.
- OpenAI. 2025. [Update to GPT-5 system card: GPT-5.2](#). Technical report, OpenAI. Accessed: 2026-01-29.
- Vardaan Pahuja, Yadong Lu, Corby Rosset, Boyu Gou, Arindam Mitra, Spencer Whitehead, Yu Su, and Ahmed Hassan. 2025. Explorer: Scaling exploration-driven web trajectory synthesis for multimodal web agents. In *Findings of ACL*, pages 6300–6323.
- Yujia Qin, Yining Ye, Junjie Fang, Haoming Wang, Shihao Liang, Shizuo Tian, Junda Zhang, Jiahao Li, Yunxin Li, Shijue Huang, Wanjuan Zhong, Kuanye Li, Jiale Yang, Yu Miao, Woyu Lin, Longxiang Liu, Xu Jiang, Qianli Ma, Jingyu Li, and 16 others. 2025. UI-TARS: Pioneering automated GUI interaction with native agents. *arXiv preprint arXiv:2501.12326*.
- Ram Ramrakhya, Andrew Szot, Omar Attia, Yuhao Yang, Anh Nguyen, Bogdan Mazouze, Zhe Gan, Harsh Agrawal, and Alexander Toshev. 2025. Scaling synthetic task generation for agents via exploration. *arXiv preprint arXiv:2509.25047*.
- Dezhi Ran, Zhi Gong, Yuzhe Guo, Mengzhou Wu, Yuan Cao, Haochuan Lu, Hengyu Zhang, Xia Zeng, Gang Cao, Liangchao Yao, Yuetang Deng, Wei Yang, and Tao Xie. 2025a. From user interface to agent interface: Efficiency optimization of UI representations for LLM agents. *arXiv preprint arXiv:2512.13438*.
- Dezhi Ran, Hao Wang, Zhihe Song, Mengzhou Wu, Yuan Cao, Ying Zhang, Wei Yang, and Tao Xie. 2024. [Guardian: A runtime framework for LLM-based UI exploration](#). In *ISSTA*, pages 958–970.
- Dezhi Ran, Hao Wang, Wenyu Wang, and Tao Xie. 2023. Badge: prioritizing UI events with hierarchical multi-armed bandits for automated UI testing. In *ICSE*, pages 894–905.
- Dezhi Ran, Mengzhou Wu, Hao Yu, Yuetong Li, Jun Ren, Yuan Cao, Xia Zeng, Haochuan Lu, Zexin Xu, Mengqian Xu, Ting Su, Liangchao Yao, Ting Xiong, Wei Yang, Yuetang Deng, Assaf Marron, David Harel, and Tao Xie. 2025b. Beyond pass or fail: Multi-dimensional benchmarking of foundation models for goal-based mobile UI navigation. *arXiv preprint arXiv:2501.02863*.
- Christopher Rawles, Sarah Clinckemaiellie, Yifan Chang, Jonathan Waltz, Gabrielle Lau, Marybeth Fair, Alice

- Li, William Bishop, Wei Li, Folawiyo Campbell-Ajala, Daniel Toyama, Robert Berry, Divya Tyamagundlu, Timothy Lillicrap, and Oriana Riva. 2025. AndroidWorld: A dynamic benchmarking environment for autonomous agents. In *ICLR*.
- Christopher Rawles, Alice Li, Daniel Rodriguez, Oriana Riva, and Timothy Lillicrap. 2023. AndroidInTheWild: A large-scale dataset for Android device control. *NeurIPS*, 36:59708–59728.
- Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. 2024. DeepSeekMath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*.
- Ting Su, Guozhu Meng, Yuting Chen, Ke Wu, Weiming Yang, Yao Yao, Guguang Pu, Yang Liu, and Zhen-dong Su. 2017. Guided, stochastic model-based GUI testing of Android apps. In *ESEC/FSE*, pages 245–256.
- Qiushi Sun, Kanzhi Cheng, Zichen Ding, Chuanyang Jin, Yian Wang, Fangzhi Xu, Zhenyu Wu, Chengyou Jia, Liheng Chen, Zhoumianze Liu, Ben Kao, Guohao Li, Junxian He, Yu Qiao, and Zhiyong Wu. 2025. OS-Genesis: Automating GUI agent trajectory construction via reverse task synthesis. In *ACL*, pages 5555–5579.
- Liujian Tang, Shaokang Dong, Yijia Huang, Minqi Xiang, Hongtao Ruan, Bin Wang, Shuo Li, Zhiheng Xi, Zhihui Cao, Hailiang Pang, Heng Kong, He Yang, Mingxu Chai, Zhilin Gao, Xingyu Liu, Yingnan Fu, Jiaming Liu, Xuanjing Huang, Yu-Gang Jiang, and 5 others. 2025. MagicGUI: A foundational mobile GUI agent with scalable data pipeline and reinforcement fine-tuning. *arXiv preprint arXiv:2508.03700*.
- Tencent. 2024. [Weixin Mini Program Platform Capabilities](#). Accessed: 2026-01-29.
- Faraz Torabi, Garrett Warnell, and Peter Stone. 2018. Behavioral cloning from observation. *arXiv preprint arXiv:1805.01954*.
- Haoming Wang, Haoyang Zou, Huatong Song, Jiazhan Feng, Junjie Fang, Juntong Lu, Longxiang Liu, Qinyu Luo, Shihao Liang, Shijue Huang, Wanjuan Zhong, Yining Ye, Yujia Qin, Yuwen Xiong, Yuxin Song, Zhiyong Wu, Aoyan Li, Bo Li, Chen Dun, and 93 others. 2025a. UI-TARS-2 technical report: Advancing GUI agent with multi-turn reinforcement learning. *arXiv preprint arXiv:2509.02544*.
- Kangrui Wang, Pingyue Zhang, Zihan Wang, Yaning Gao, Linjie Li, Qineng Wang, Hanyang Chen, Yiping Lu, Zhengyuan Yang, Lijuan Wang, Ranjay Krishna, Jiajun Wu, Fei-Fei Li, Yejin Choi, and Manling Li. 2025b. [VAGEN: Reinforcing world model reasoning for multi-turn VLM agents](#). In *NeurIPS*, pages 172871–172933.
- Shuai Wang, Weiwen Liu, Jingxuan Chen, Yuqi Zhou, Weinan Gan, Xingshan Zeng, Yuhan Che, Shuai Yu, Xinlong Hao, Kun Shao, Bin Wang, Chuhan Wu, Yasheng Wang, Ruiming Tang, and Jianye Hao. 2024. GUI agents with foundation models: A comprehensive survey. *arXiv preprint arXiv:2411.04890*.
- Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc Le, and Denny Zhou. 2022. Chain-of-thought prompting elicits its reasoning in large language models. *NeurIPS*, 35:24824–24837.
- Mengzhou Wu, Yuzhe Guo, Yuan Cao, Haochuan Lu, Hengyu Zhang, Xia Zeng, Liangchao Yao, Yuetang Deng, Dezhi Ran, Wei Yang, and Tao Xie. 2025. [Element-aware fine-tuning of vision-language models for cost-efficient GUI testing in an industrial setting](#). In *ASE*, pages 3581–3592.
- Mengzhou Wu, Hao Wang, Jun Ren, Yuan Cao, Yuetong Li, Alex Jiang, Dezhi Ran, Yitao Hu, Wei Yang, and Tao Xie. 2024a. Skill-adaptive imitation learning for UI test reuse. *arXiv preprint arXiv:2409.13311*.
- Zhiyong Wu, Zhenyu Wu, Fangzhi Xu, Yian Wang, Qiushi Sun, Chengyou Jia, Kanzhi Cheng, Zichen Ding, Liheng Chen, Paul Pu Liang, and Yu Qiao. 2024b. OS-ATLAS: A foundation action model for generalist GUI agents. *arXiv preprint arXiv:2410.23218*.
- Yifan Xu, Xiao Liu, Xinghan Liu, Jiaqi Fu, Hanchen Zhang, Bohao Jing, Shudan Zhang, Yuting Wang, Wenyi Zhao, and Yuxiao Dong. 2025a. MobileRL: Online agentic reinforcement learning for mobile GUI agents. *arXiv preprint arXiv:2509.18119*.
- Yiheng Xu, Zekun Wang, Junli Wang, Dunjie Lu, Tianbao Xie, Amrita Saha, Doyen Sahoo, Tao Yu, and Caiming Xiong. 2025b. [Aguvis: Unified pure vision agents for autonomous GUI interaction](#). In *ICML*, pages 69772–69805.
- Jiabo Ye, Xi Zhang, Haiyang Xu, Haowei Liu, Junyang Wang, Zhaoqing Zhu, Ziwei Zheng, Feiyu Gao, Junjie Cao, Zhengxi Lu, Jitong Liao, Qi Zheng, Fei Huang, Jingren Zhou, and Ming Yan. 2025. Mobile-Agent-v3: Fundamental agents for GUI automation. *arXiv preprint arXiv:2508.15144*.
- Jinghan Yu, Junhao Xiao, Zhiyuan Ma, Yue Ma, Kaiqi Liu, Yuhang Wang, Daizong Liu, Xianghao Meng, and Jianjun Li. 2025. MILD: Multi-layer diffusion strategy for complex and precise multi-IP aware human erasing. *arXiv preprint arXiv:2508.06543*.
- Jinghan Yu, Junhao Xiao, Chenyu Zhu, Jiaming Li, Jia Li, HanMing Deng, Xirui Wang, Guoli Jia, Jianjun Li, Xiang Bai, Bowen Zhou, and Zhiyuan Ma. 2026. [I2E: From image pixels to actionable interactive environments for text-guided image editing](#). In *ACL*, pages 44827–44850.
- Yang Yue, Zhiqi Chen, Rui Lu, Andrew Zhao, Zhaokai Wang, Shiji Song, and Gao Huang. 2025. Does

- reinforcement learning really incentivize reasoning capacity in LLMs beyond the base model? *arXiv preprint arXiv:2504.13837*.
- Charlie Zhang, Graham Neubig, and Xiang Yue. 2025a. On the interplay of pre-training, mid-training, and RL on reasoning language models. *arXiv preprint arXiv:2512.07783*.
- Danyang Zhang, Situo Zhang, Ziyue Yang, Zichen Zhu, Zihan Zhao, Ruisheng Cao, Lu Chen, and Kai Yu. 2025b. ProgRM: Build better GUI agents with progress rewards. *arXiv preprint arXiv:2505.18121*.
- Jiwen Zhang, Jihao Wu, Yihua Teng, Minghui Liao, Nuo Xu, Xiao Xiao, Zhongyu Wei, and Duyu Tang. 2024. Android in the zoo: Chain-of-Action-Thought for GUI agents. *arXiv preprint arXiv:2403.02713*.
- Kai Zhang, Xiangchao Chen, Bo Liu, Tianci Xue, Zeyi Liao, Zhihan Liu, Xiyao Wang, Yuting Ning, Zhaorun Chen, Xiaohan Fu, Jian Xie, Yuxuan Sun, Boyu Gou, Qi Qi, Zihang Meng, Jianwei Yang, Ning Zhang, Xian Li, Ashish Shah, and 11 others. 2025c. Agent learning via early experience. *arXiv preprint arXiv:2510.08558*.

A Problem Formulation

We formulate GUI navigation not merely as trajectory execution but as learning the underlying transition dynamics of the interface.

GUI as a State Transition Graph. We model the GUI environment as a directed state transition graph $\mathcal{G} = (\mathcal{S}, \mathcal{E})$. Each node $s \in \mathcal{S}$ represents a unique UI state, formally defined as a tuple $s = (I, X)$, where $I \in \mathbb{R}^{H \times W \times 3}$ is the pixel-level screenshot and X denotes structural metadata (e.g., the accessibility tree). A directed edge $e \in \mathcal{E}$ represents an atomic transition (s_t, a_t, s_{t+1}) . Here, an action $a_t \in \mathcal{A}$ is defined as a tuple $a_t = (\tau, \theta)$, where τ denotes the operation type (e.g., *click*, *scroll*, *input*) and θ represents the corresponding parameters (e.g., screen coordinates or input text content), with the full specification of $\mathcal{A}$ detailed in Appendix C. Executing a_t in state s_t leads to state s_{t+1} .

Forward Dynamics (Outcome Prediction). Forward prediction serves as a training objective for a semantic transition world model by predicting the next state s_{t+1} given the current state s_t and action a_t . We formulate this objective by minimizing the following negative log-likelihood.

$$\mathcal{L}_{\text{fwd}} = -\mathbb{E}_{(s_t, a_t, s_{t+1}) \sim \mathcal{D}} [\log P_{\theta}(s_{t+1} \mid s_t, a_t)] \quad (1)$$

Inverse Dynamics (Action Inference). Inverse dynamics aims to infer the action a_t responsible for the transition between s_t and s_{t+1} . The loss function is defined as follows.

$$\mathcal{L}_{\text{inv}} = -\mathbb{E}_{(s_t, a_t, s_{t+1}) \sim \mathcal{D}} [\log P_{\theta}(a_t \mid s_t, s_{t+1})] \quad (2)$$

Backward Dynamics (Precondition Inference). Complementary to forward dynamics, this objective learns to infer the precondition state s_t given

the executed action a_t and the outcome s_{t+1} . We minimize the following objective.

$$\mathcal{L}_{\text{bwd}} = -\mathbb{E}_{(s_t, a_t, s_{t+1}) \sim \mathcal{D}} [\log P_{\theta}(s_t \mid s_{t+1}, a_t)] \quad (3)$$

B Extended Discussion on Related Work

This appendix extends the discussion of background and related work in Section 2.

B.1 Learning Paradigms for GUI Agents

Recent advances in generalist GUI agents have primarily focused on refining policy optimization paradigms. While initial efforts rely on imitation learning, often through Behavioral Cloning (BC) (Wu et al., 2024b; Lin et al., 2025a; Lu et al., 2025; Chen et al., 2025; Wu et al., 2024a), the field has rapidly shifted toward multi-step Reinforcement Learning (RL) (Ye et al., 2025; Xu et al., 2025a; Wang et al., 2025a) integrated with Chain-of-Thought (CoT) (Wei et al., 2022) reasoning to handle non-stationary, long-horizon tasks. However, recent studies on RL reveal a critical bottleneck: RL alone struggles to surpass the inherent capability ceiling of the base model (Zhang et al., 2025a; Yue et al., 2025). This bottleneck underscores the necessity of elevating the foundation model’s intrinsic capabilities via Continual Pre-Training (CPT) prior to policy alignment. Existing CPT methods for GUI agents, however, are largely confined to static UI understanding, such as element grounding or screen captioning (Tang et al., 2025; Wu et al., 2024b). While these methods improve perception, they fail to capture the interaction dynamics of the interface. Distinct from these static objectives, our work targets a dynamic objective: learning a GUI transition world model via CPT on atomic transitions. By grounding the model in the forward dynamics of the interface (i.e., predicting consequences), we provide a scalable path to learning transition-aware representations for generalist GUI agents.

Table 6: Definition of the Action Space.

Type (τ)	Params (θ)	Description
CLICK	(x, y)	Tap at pixel or normalized coordinates.
INPUT	(x, y, text)	Focus on the target element at (x, y) and type the provided text string.
SCROLL	(x, y, dir)	Scroll from (x, y) toward UP, DOWN, LEFT, or RIGHT.
FINISH	$\emptyset$	Terminate the current episode immediately.

B.2 Scalable Data Acquisition for GUI Agents

A fundamental challenge in training generalist GUI agents is the data scalability bottleneck of Behavioral Cloning (BC). Practically, collecting high-quality human demonstrations is prohibitively expensive. To bypass these limitations, the field has pivoted toward synthetic data generation, primarily following two directions. One line of work attempts to synthesize execution trajectories by deploying agents to interact with the environment under specific instructions (Lin et al., 2025b; Sun et al., 2025; Ramrakhya et al., 2025; Pahuja et al., 2025). However, due to the strict long-horizon dependencies of GUI tasks, generating valid data is notoriously difficult; a single intermediate error invalidates the entire trajectory, resulting in extremely low yields of high-quality samples. Alternatively, other methods focus on static tasks such as screen captioning or VQA by leveraging powerful foundation models to label GUI screenshots (Qin et al., 2025; Wu et al., 2024b; Tang et al., 2025). Yet, these methods rely on model-based distillation, increasing the risk that an agent inherits teacher-model hallucinations and lacks objective grounding in real execution. In contrast, our **UI-OCEANUS** framework reduces reliance on ungrounded teacher supervision by grounding supervision in observed state transitions. By treating state transitions as an execution-based verification signal, we mitigate the fragility of long-horizon synthesis and the bias of distillation, enabling performance to scale with autonomous exploration.

B.3 World Models and Dynamics in Digital Interfaces

Conceptualizing the GUI as a dynamic system aligns with the broader literature on world models in model-based reinforcement learning and representation learning, where frameworks (Hafner

et al., 2025; Assran et al., 2023; LeCun, 2022) enable agents to plan by modeling future latent states. Instruction-guided image editing similarly maps an input visual state and an editing instruction to a transformed visual state (Brooks et al., 2023; Huang et al., 2024). Recent work extends this paradigm toward structured and localized transformations, including multi-layer human erasing (Yu et al., 2025) and action-based editing within interactive object-layer environments (Yu et al., 2026). In the specific domain of GUIs, prior work such as ViMo (Luo et al., 2025) and VAGEN (Wang et al., 2025b) has explored this direction through explicit visual state prediction, while GUI-Shift (Gao et al., 2026) uses an inverse dynamics objective to infer actions from state changes. Recent concurrent work (Zhang et al., 2025c) uses implicit world modeling as an auxiliary task but is limited to local branching from expert demonstrations, lacking a scalable, task-agnostic Continual Pre-Training objective for learning broad transition patterns. We distinguish our work by leveraging autonomous exploration and intrinsic environmental feedback to learn a GUI transition world model via Continual Pre-Training, establishing a transition-aware representation before agentic post-training.

C Action Space Specification

We define the action space $\mathcal{A}$ as a set of atomic operations supported by an agent. Each action is represented as a tuple $a_t = (\tau, \theta)$, where τ specifies the primitive type and θ contains the necessary parameters. The representation of spatial coordinates (x, y) adapts to the underlying vision-language backbone:

- **Qwen2.5-VL series** uses **absolute pixel coordinates** $(x, y) \in [0, W] \times [0, H]$ based on the original screenshot resolution.
- **Qwen3-VL series** uses **normalized coordinates** quantized to the integer range $[0, 1000]$.

The specific primitives and their parameter structures are summarized in Table 6.

D Details of the Data Filtering Pipeline

This section details the implementation of the deduplication stages using Locality-Sensitive Hashing (LSH) for both graph-structured accessibility trees and high-dimensional screenshots.

Table 7: **Action Distribution before and after Filtering.**

Action	Raw	Filtered
Click	60.2%	71.6%
Input	7.0%	12.0%
Scroll	5.0%	9.9%
Back	27.8%	6.5%

Table 8: **Semantic Filtering Breakdown.** Percentages are measured within the semantic filtering stage.

Category	Disposition	Share
Loading screens	Removed	1.2%
Error screens	Removed	0.1%
Latency no-ops	Removed	16.3%
Auto-play changes	Removed	1.9%
Popups	Removed	1.4%
Aligned static transitions	Retained	8.1%

D.1 Structural Transition Deduplication

To identify interface states with identical logic but varying content, we model transitions as sets of structural tokens.

Tokenization & MinHash. We extract tokens from the accessibility tree nodes based on semantic attributes (e.g., tag, xpath, events). A transition is represented by the union of hashed tokens from both the pre- and post-action states. We generate MinHash signatures to efficiently estimate Jaccard similarity, reducing variable-length token sets to fixed-length vectors.

LSH Indexing. We use a banding strategy to index these signatures. Transitions colliding in any band are retrieved as candidates. We then verify candidates against a Jaccard similarity threshold, retaining only unique representatives for the final dataset.

D.2 Visual Transition Deduplication

For opaque components (e.g., WebViews), we use a composite visual fingerprinting strategy combining frequency and gradient domains.

Composite Hashing. For each transition, we concatenate the **Perceptual Hash (pHash)** and **Difference Hash (dHash)** of the start and end screen-

shots. This combination ensures robustness against rendering noise while remaining sensitive to significant layout changes.

Clustering Strategy. We use bit-sampling LSH to project high-dimensional hashes into lower-dimensional buckets for indexing. Candidates are clustered using a Union-Find data structure based on Hamming distance thresholds. For each connected component, the transition with the highest data source priority is retained.

D.3 Semantic Filtering

We use Qwen3-VL-235B-Instruct as a semantic verifier. The model is prompted to output a binary validity score, retaining only transitions where the screen changes logically correspond to the annotated action.

D.4 Filtering Analysis

We further analyze how filtering changes the transition corpus. As shown in Table 7, filtering concentrates high-information actions such as clicks, inputs, and scrolls, while pruning a large portion of repetitive back navigation.

State Coverage. We track distinct path keys, a signal not used by the filters, to estimate state coverage changes. The total number of distinct path keys drops by only 4.3%, suggesting that filtering removes redundancy without collapsing macro-level state diversity. Rare paths with 1–5 occurrences retain 60% of their volume, while highly repetitive paths with more than 1000 occurrences drop to 4%, indicating that the pipeline primarily removes frequent duplicates.

Semantic Filtering Breakdown. Table 8 summarizes the semantic filtering stage. The removed cases are dominated by valid but action-disconnected events such as latency no-ops, rather than invalid screens. We also explicitly retain causally aligned but visually unchanged transitions to avoid removing interaction-relevant cases that are not visually salient.

E Training Dynamics Analysis

Figure 4 illustrates the training dynamics discussed in Section 4.3. The faster drop in the inverse-dynamics loss is consistent with its sparse action target being easier to fit, while the forward-dynamics loss remains higher for longer. The curve diagnoses objective difficulty rather than explaining the downstream gap; Appendix F provides target-density controls.

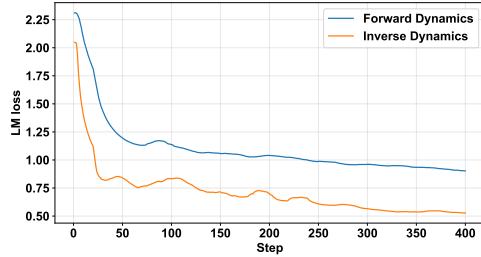


Figure 4: **Training Loss Comparison.** Inverse dynamics (orange) converges faster under sparse action targets, while forward dynamics (blue) maintains a higher loss level for longer.

F Controlled Objective Comparisons

Table 9 reports additional controls for the objective comparison in Section 4.3. The dense rich-inverse objective augments inverse dynamics with synthesized CoT rationales generated in hindsight by Qwen3.5-VL-397B-A17B, making its output density comparable to forward and backward dynamics. These controls help disentangle temporal direction, output modality, and target-token volume.

Output Richness. Rich inverse dynamics controls for target richness by replacing sparse action labels with dense CoT-and-action outputs. The dense target improves over standard inverse dynamics in average EM (54.6 versus 48.7), but still trails full-scale forward dynamics (56.4). This result suggests that the advantage of future-state prediction is not solely explained by richer textual outputs.

Temporal Direction. Backward dynamics uses a dense state-description target comparable to forward dynamics, but predicts the past state rather than the future state. Forward dynamics remains stronger than backward dynamics at the same data scale in average EM (56.4 versus 54.6). This result indicates that action-conditioned future prediction is the most effective objective among the variants tested.

Target-Token Volume. The 12.5% forward setting provides a coarse control for total target-token volume: $12.5\% \times 80 \approx 10$ tokens per original transition, matching the approximate token budget of standard inverse dynamics. Even under this reduced data scale, forward dynamics reaches an average EM of 53.9, exceeding standard inverse dynamics at 48.7. This control does not remove all differences in target content and structure, but it suggests that the gain is not simply due to the total number of target tokens.

G Teacher Model Ablation

UI-OCEANUS uses a VLM as a linguistic interpreter of observed transitions rather than as a generator of transitions. Both the pre-state and post-state screenshots are produced by actual system execution, while the teacher verbalizes the observed transition into text supervision. To study sensitivity to teacher scale, we conduct a side-by-side preference study on 1,000 randomly sampled transitions. For each transition, we compare a description generated by a challenger teacher with one generated by Qwen3-VL-235B-A22B, randomize the candidate order, and ask an independent Qwen3.5-397B-A17B judge to choose the better description or declare a tie.

The results show mid-scale saturation: Qwen3-VL-32B outperforms the 235B interpreter in this preference study, while the larger 397B model does not yield further gains. This finding supports a more modest interpretation of the teacher role. The supervision remains language-mediated and therefore still depends on an interpreter, but the core transition signal is grounded in executed UI state changes rather than generated by the teacher itself. Similarly, semantic filtering removes only about 21% of post-deduplication transitions and checks action-feedback consistency, a simpler task than producing the transition. We therefore revise the claim from eliminating teacher dependence to reducing reliance on extreme-scale teacher capabilities by grounding supervision in observed transitions.

H Hyperparameter Settings

We detail the hyperparameter configurations for both the Continual Pre-Training (CPT) phase and the Agentic Post-Training phase.

Table 9: **Target-Density Controls for Dynamics Objectives.** The controlled objectives are evaluated on Qwen3-VL-2B and Qwen3-VL-4B. Standard inverse, backward, and 100% forward dynamics correspond to the main ablation; 12.5% forward dynamics is included as a target-token-volume control.

Objective	Target Type	Density	Tokens	Scale	2B EM	4B EM	Avg EM
Std. Inv.	Action	Sparse	~ 10	100%	46.4	50.9	48.7
Rich Inv.	CoT+Action	Dense	~ 100	100%	49.5	59.6	54.6
Backward	Past State	Dense	~ 80	100%	50.4	58.7	54.6
Forward	Future State	Dense	~ 80	12.5%	49.4	58.3	53.9
Forward	Future State	Dense	~ 80	100%	52.7	60.0	56.4

Table 13: **General Benchmark Preservation.** Positive values indicate improvements after **UI-OCEANUS CPT**.

Benchmark	Base	Ours	Δ
MMBench (EN)	86.46	83.36	-3.10
MMStar	67.93	67.20	-0.73
MMMU (dev)	57.30	59.30	+2.00
MMMU (val)	61.90	60.00	-1.90
MathVista	74.70	72.80	-1.90
AI2D	82.93	83.52	+0.59
OCRBench	832	839	+7
MMVet	78.07	77.34	-0.73
TextVQA	77.34	80.21	+2.87
DocVQA	90.35	91.70	+1.35
MMLU	79.45	79.17	-0.28
C-Eval	80.89	81.63	+0.74

Table 10: **Teacher Model Preference Study.** Percentages are measured over 1,000 randomly sampled transitions. Candidate order is randomized.

Challenger	Challenger Wins	Tie	235B Wins
Qwen3-VL-8B	15.6%	43.0%	41.4%
Qwen3-VL-32B	38.2%	43.1%	18.7%
Qwen3.5-397B-A17B	22.6%	49.0%	28.4%

H.1 Continual Pre-Training (CPT)

The hyperparameters for the CPT phase are summarized in Table 11.

Table 11: **Hyperparameters for Continual Pre-Training.**

Hyperparameter	32B Models	7B–8B Models	2B–4B Models
Optimizer		AdamW	
Global Batch Size	2048	1024	1024
Peak Learning Rate	1×10^{-5}	2×10^{-5}	3×10^{-5}
LR Scheduler		Cosine	
Warmup Ratio		0.03	
Max Sequence Length		4096	
Weight Decay	0.10	0.01	0.01
Epochs		1	

H.2 Agentic Post-Training: SFT

Table 12 lists the settings for the Supervised Fine-Tuning stage.

Table 12: **Hyperparameters for SFT.**

Hyperparameter	32B Models	7B–8B Models	2B–4B Models
Optimizer		AdamW	
Global Batch Size		64	
Peak Learning Rate	3×10^{-6}	6×10^{-6}	1×10^{-5}
LR Scheduler		Constant	
Max Sequence Length		4096	
Weight Decay		0	
Epochs		2	

I General Capability Preservation

To assess whether GUI-specialized CPT harms general capabilities, we compare the Qwen2.5-VL-32B base model with its **UI-OCEANUS CPT** counterpart on 11 public multimodal and reasoning benchmarks, with MMMU reported separately on dev and val splits. As shown in Table 13, general performance is largely preserved, with mild regressions on several tasks and gains on structured visual understanding benchmarks such as TextVQA and DocVQA.

J Dataset Statistics and SOTA Comparisons

We provide detailed statistics for the datasets used in our experiments, alongside a performance comparison against state-of-the-art proprietary models to assess the ceiling of our framework.

J.1 Offline Scalability Benchmark

Data Distribution. Table 14 summarizes the distribution of action primitives in both the SFT training set and the held-out test set used in Section 4.1. The SFT training set reflects the natural frequency of user interactions (dominated by Click, 79.0%), whereas the evaluation benchmark is curated to ensure balanced coverage, specifically increasing the representation of Scroll (9.8%) and Input (11.0%) to rigorously test generalization.

Table 14: **Statistics of Action Primitives (Offline).** Counts and percentages of action types in the SFT training and test sets.

Action Type	SFT Training Set		Test Set	
	Count	Pct.	Count	Pct.
Click	6,229	79.0%	6,048	69.6%
Input	608	7.7%	954	11.0%
Finish	944	12.0%	839	9.6%
Scroll	105	1.3%	849	9.8%
Total	7,886	100.0%	8,690	100.0%

Comparison with Proprietary Models (Offline).

Table 15 presents the performance comparison between our **UI-OCEANUS** framework and state-of-the-art proprietary models (including GPT-5.2, Claude-4.5-Opus, and Gemini-3-Flash) as well as large-scale open-weight baselines on the offline scalability benchmark.

J.2 Online Evaluation Benchmark

Complementing the offline metrics, the online benchmark used in Section 4.2 evaluates generalization in dynamic environments.

Diversity and Coverage. The benchmark comprises 149 instructions covering over 50 unique mini-programs (Table 16). It spans seven domains, from high-frequency scenarios (e.g., Food) to long-tail services (e.g., Medical), supporting evaluation on transferable interaction patterns rather than memorized layouts.

Comparison with Proprietary Models (Online).

Table 17 presents the success rate (SR) on the online benchmark. We report the performance of **UI-OCEANUS** across three progressive post-training stages and compare it with representative external models, including state-of-the-art proprietary models.

Table 15: **Offline Benchmark Performance Compared with Proprietary & Large-Scale Models.** We report Exact Match (EM) and Type Match (TM) scores across different model families.

Model	Exact Match (EM)	Type Match (TM)
<i>Proprietary Models</i>		
GPT-5.2 (OpenAI, 2025) (w/ Grounding)	61.1	76.9
Claude-4.5-Sonnet (Anthropic, 2025b) (w/ Grounding)	64.8	81.2
Claude-4.5-Opus (Anthropic, 2025a) (w/ Grounding)	69.2	82.7
Gemini-3-Flash (Google, 2025) (w/ Grounding)	69.2	82.0
<i>Open-Weight Baselines</i>		
Qwen2.5-VL-32B	33.9	56.6
Qwen2.5-VL-72B	51.4	77.2
Qwen3-VL-32B	47.1	79.2
Qwen3-VL-235B-A22B	47.5	79.4
Qwen2.5-VL-32B (w/ Grounding)	56.5	77.9
Qwen2.5-VL-72B (w/ Grounding)	56.8	77.2
Qwen3-VL-32B (w/ Grounding)	61.2	79.3
Qwen3-VL-235B-A22B (w/ Grounding)	61.5	79.3
<i>Ours</i>		
Qwen3-VL-32B + UI-OCEANUS (CPT+SFT)	65.0	81.1
Qwen2.5-VL-32B + UI-OCEANUS (CPT+SFT)	64.8	82.1
Qwen2.5-VL-32B + UI-OCEANUS (CPT+SFT+GRPO)	68.9	84.5

Table 16: **Domain Distribution of the Online Benchmark.** The benchmark covers eight distinct categories and over 50 unique mini-programs.

Domain	Tasks	Representative mini-programs
Travel & Transport	25	Ctrip, Didi, Railway 12306
Food & Drink	21	Starbucks, KFC, Meituan, Heytea
Entertainment & Sports	15	Maoyan, Damai, Supermonkey, Golflive
Shopping & Logistics	13	JD.com, SF Express, Xianyu, Cainiao
Life Services	13	State Gov. Service, Beike, China Mobile
Medical & Health	11	Tencent Health
Office & Efficiency	8	Tencent Docs, Wenjuanxing, Tencent Meeting
<i>Others</i>	43	Diverse long-tail utilities
Total	149	over 50 unique mini-programs

Table 17: **Online Success Rate Compared with Proprietary Models.** Comparison of Success Rate (SR) on live mini-programs between our progressive pipeline and external baselines.

Category	Model / Stage	SR (%)	Gap w/ Best Ours
<i>Ours (32B)</i>	UI-OCEANUS (SFT Cold Start)	29.5	-14.8
	+ Step-level GRPO	38.9	-5.4
	+ Multi-step GRPO	44.3	—
<i>Proprietary & External Baselines</i>	Seed-1.8 (ByteDance Seed, 2025) (w/ Grounding)	27.5	-16.8
	Claude-4.5-Opus (Anthropic, 2025a) (w/ Grounding)	46.3	+2.0
	Gemini-3-Flash (Google, 2025) (w/ Grounding)	49.0	+4.7