

# UI-AGILE: Advancing GUI Agents with Effective Reinforcement Learning and Precise Inference-Time Grounding

## Supplementary Material

### A. Environment and Implementation Details

In the following, we provide the details of experiment environment and our implementation.

**Environment.** We use a machine with two Intel(R) Xeon(R) Silver 4314 CPU @ 2.40GHz, 512GB main memory and eight NVIDIA A800 GPU for experiments.

**Training Details.** We use the trl framework<sup>1</sup> to implement the cropping-based resampling strategy and reward functions. The sampling process is attempted 4 times at most and is bypassed entirely if the bbox's dimensions exceed the target crop size. Following prior works, we use Qwen2.5-VL-3B<sup>2</sup> and Qwen2.5-VL-7B<sup>3</sup> as base models.

**Inference Details.** For decomposed grounding with selection, the input image is divided into four sub-images scaling to 60% of the original dimensions, with adjacent sub-images overlapping by 10% of the original image's width and height. In the element image extraction stage, we define the element's area by creating a simple bounding box centered on the predicted point with the width and height equal to 14% of the sub-image's width and height. We have also explored a more sophisticated approach using OmniParser to refine this bounding box. However, it does not improve performance and increases the inference overhead. In the selection stage, we use Qwen2.5VL-7B-instruct to choose the final answer and the prompt is listed in Fig. 5.

**Hyperparameters.** Tab. 4 provides the training hyperparameters of UI-AGILE where cropping factor is the width and height ratio of new attempted image and last attempted image. The hyperparameters in Eq. 2 and Eq. 5 are optimized using a grid search. Tab. 5 shows the specific values.

### B. Code

We provide the code for our RFT training and the Decomposed Grounding with Selection method in two separate modules. To avoid potential dependency conflicts, each module is designed to be run in its own conda environment.

To ensure a fair and comprehensive comparison, we conducted extensive experiments on the ScreenSpot-Pro benchmark. This involved re-implementing baseline models and

<sup>1</sup><https://github.com/huggingface/trl>

<sup>2</sup><https://huggingface.co/Qwen/Qwen2.5-VL-3B-Instruct>

<sup>3</sup><https://huggingface.co/Qwen/Qwen2.5-VL-7B-Instruct>

Table 4. Training hyperparameters.

Hyperparameter	Value
learning rate	from 1e-06 to 4.36e-10
num generations	8
num train epochs	2
per device train batch size	4
gradient accumulation steps	4
cropping factor	0.6
sampling attempt num	4

Table 5. Hyperparameters for Eq. 2 and Eq. 5

Hyperparameter	Value
$l_{\text{ideal\_start}}$	120 chars
$l_{\text{ideal\_end}}$	200 chars
$l_{\text{min}}$	50 chars
$l_{\text{max}}$	300 chars
$C$	4

### Prompt for VLM-based Adjudication

**Instruction:** {instruction content}.

**Question:** Does this image accurately match the instruction? Yes or No?

**Answer:**

Figure 5. Prompt for VLM-based adjudication.

evaluating them with our Decomposed Grounding with Selection method.

We use the parquet format to store test data in order to reduce the I/O read overhead.

We still have room for improvement in the implementation of Decomposed Grounding with Selection for Inference, including multi-threading to accelerate processing images and other non-GPU operations.

### C. Additional Results for Ablation Study

Fig. 6 provides results of the ablation study using Type and GR as evaluation metrics on AndroidControl. We can observe that each component in UI-AGILE indeed contributes to the overall performance.

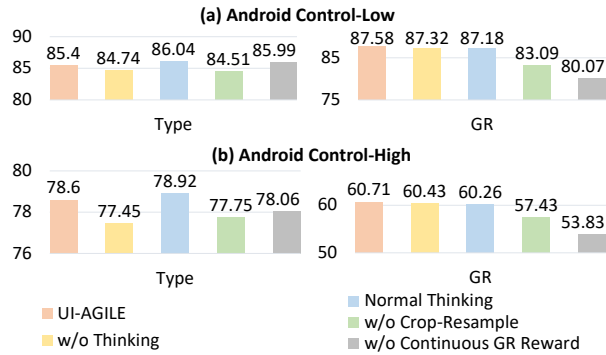


Figure 6. Ablation study using Type and GR as evaluation metrics on AndroidControl.

## D. Analysis of Inference Time

We report the inference time of our decomposed grounding with selection method on the full ScreenSpot-Pro dataset [12] using the vLLM framework [10] and one 80G A800 GPU card.

As a baseline, the standard grounding approach applied to UI-AGILE-7B completes the benchmark in **30 minutes**. When applying our method, the decomposed grounding stage takes **35 minutes**. The subsequent VLM-based selection stage requires additional **4 minutes**. The modest increase in overhead is a practical trade-off for the substantial gain of grounding accuracy brought by our method.