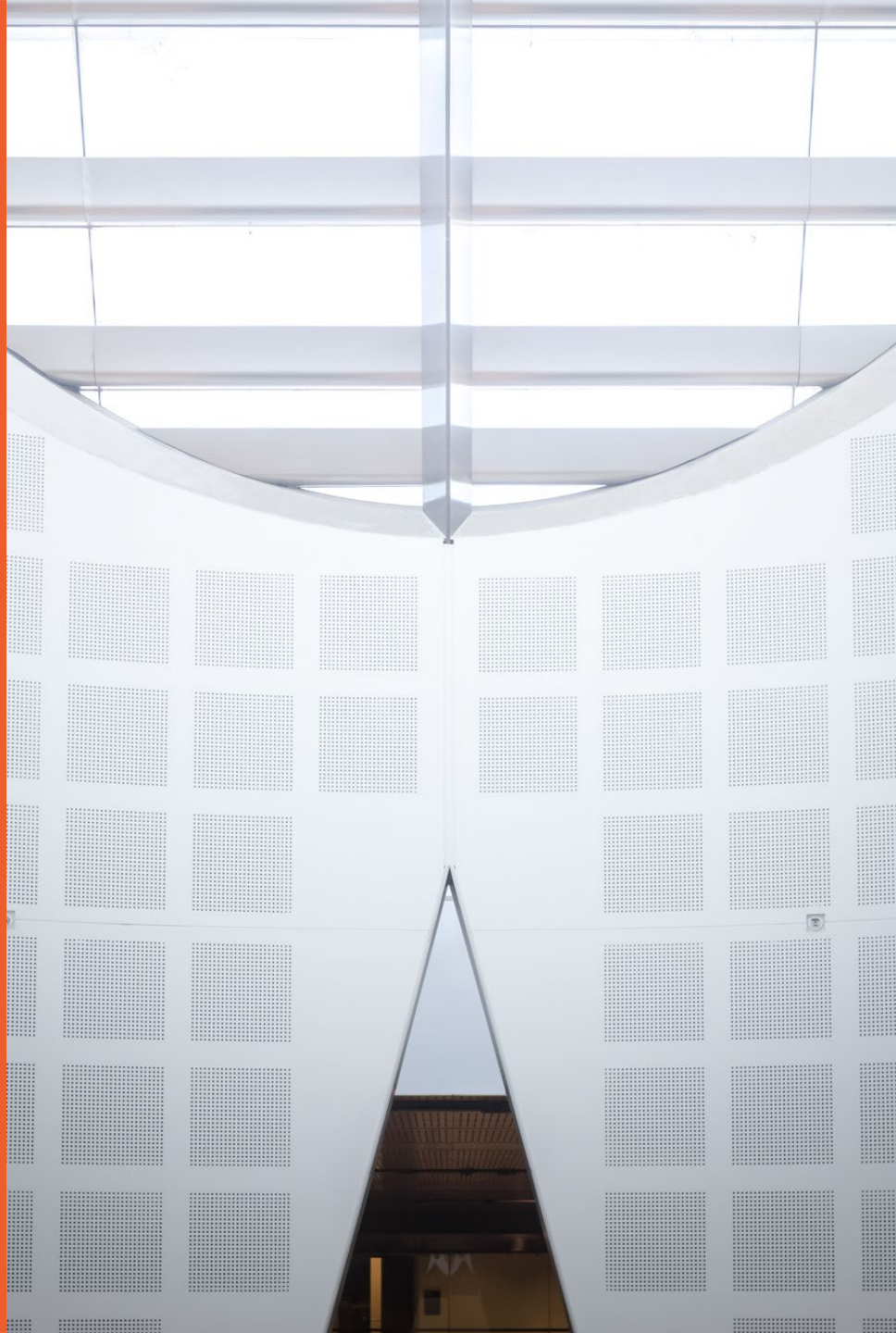
AccuAnnotate

Teaching AI to click precisely on desktop screens

Presented by
Ning Bao
Faculty of Engineering
School of Computer Science

THE UNIVERSITY OF SYDNEY



Introduction

"AccuAnnotate: Pixel-Accurate Graphical User Interface Annotation and Reinforcement Learning Fine-Tuning of a Compact Vision–Language Model for Desktop Grounding"

THE UNIVERSITY OF SYDNEY

What is “grounding”?

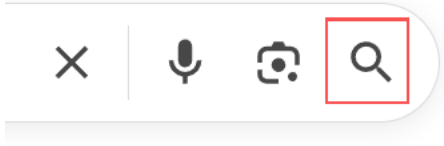
- *Grounding* = connecting **language** to the **right place or object** in the world.

- **In this thesis:**

  “Given a text instruction and a screenshot, grounding means finding the exact pixel to click on the screen.”

- Pixel-level grounding → the click isn’t just “roughly here”, it must be precise inside the target UI element.

Click the ‘Search’ icon →



What is a Vision–Language Model (VLM)?

- A **Vision–Language Model** takes images + text as input and produces text (or numbers) as output.

- It is pre-trained on large datasets of image–caption pairs
  - (e.g. “a button labelled Save on a toolbar”).

- For GUIs, it sees a screenshot and a natural language instruction, and must reason about both.

The big picture: why GUI grounding?

What a GUI agent needs to do:

- "see" the screen, understand the instruction, click the right pixel.

Why it's hard:

- Interfaces are dense and diverse (menus, icons, toolbars).
- Agents must be precise (safety, reliability).

Problems today:

- Lots of mobile + web data, but few pixel-accurate desktop datasets.
- Auto-labels are often noisy or coarse.

Research Question

Problem:

- Given a desktop screenshot + natural language instruction, predict a click point on the screen (pixel-level grounding).

Research question:

- "How can we automatically produce high-fidelity, pixel-accurate annotations for desktop GUI grounding, and to what extent do these datasets enable reinforcement learning fine-tuning of a compact VLM to improve grounding performance?"

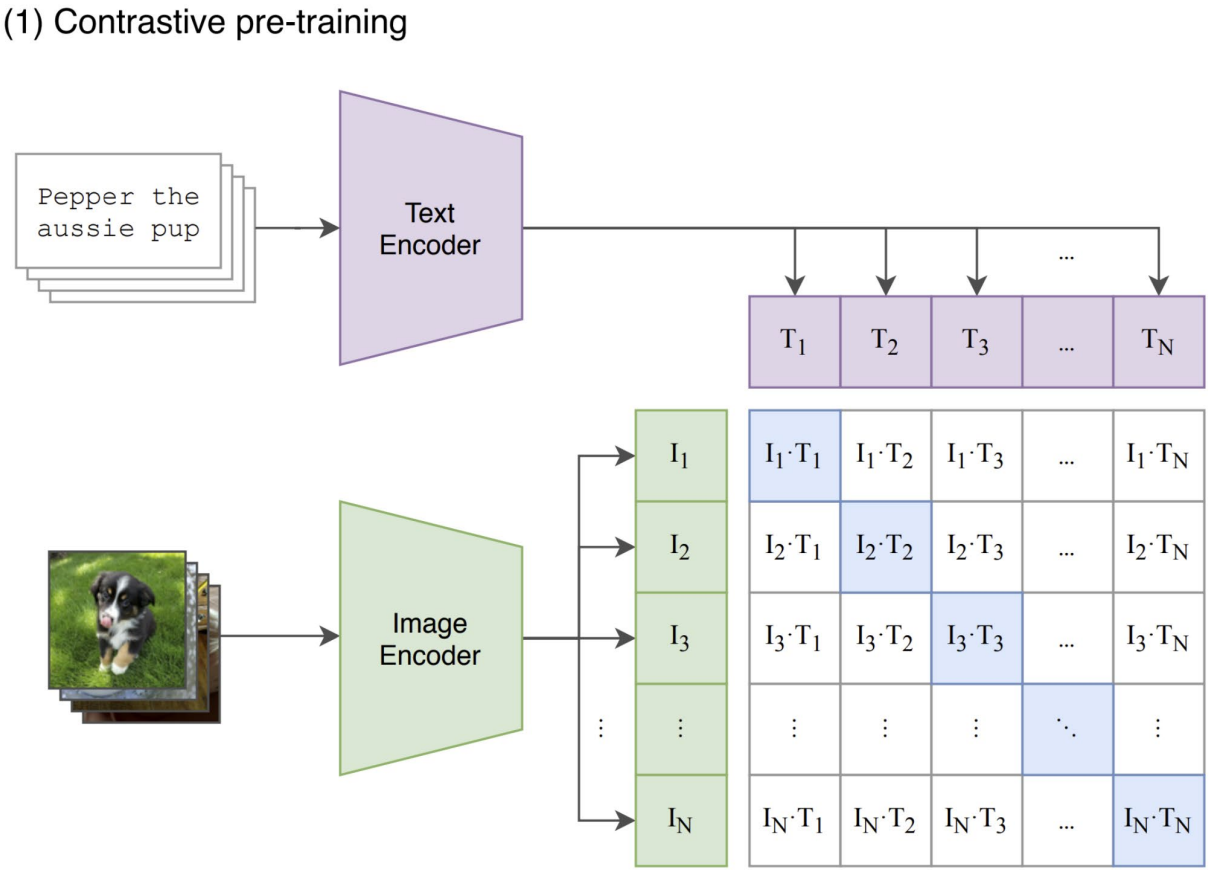
Literature Review

THE UNIVERSITY OF SYDNEY

General Purpose VLMs

CLIP [1]

- Contrastive training on 400 M image–text pairs → shared embeddings



General Purpose VLMs

CLIP [1]

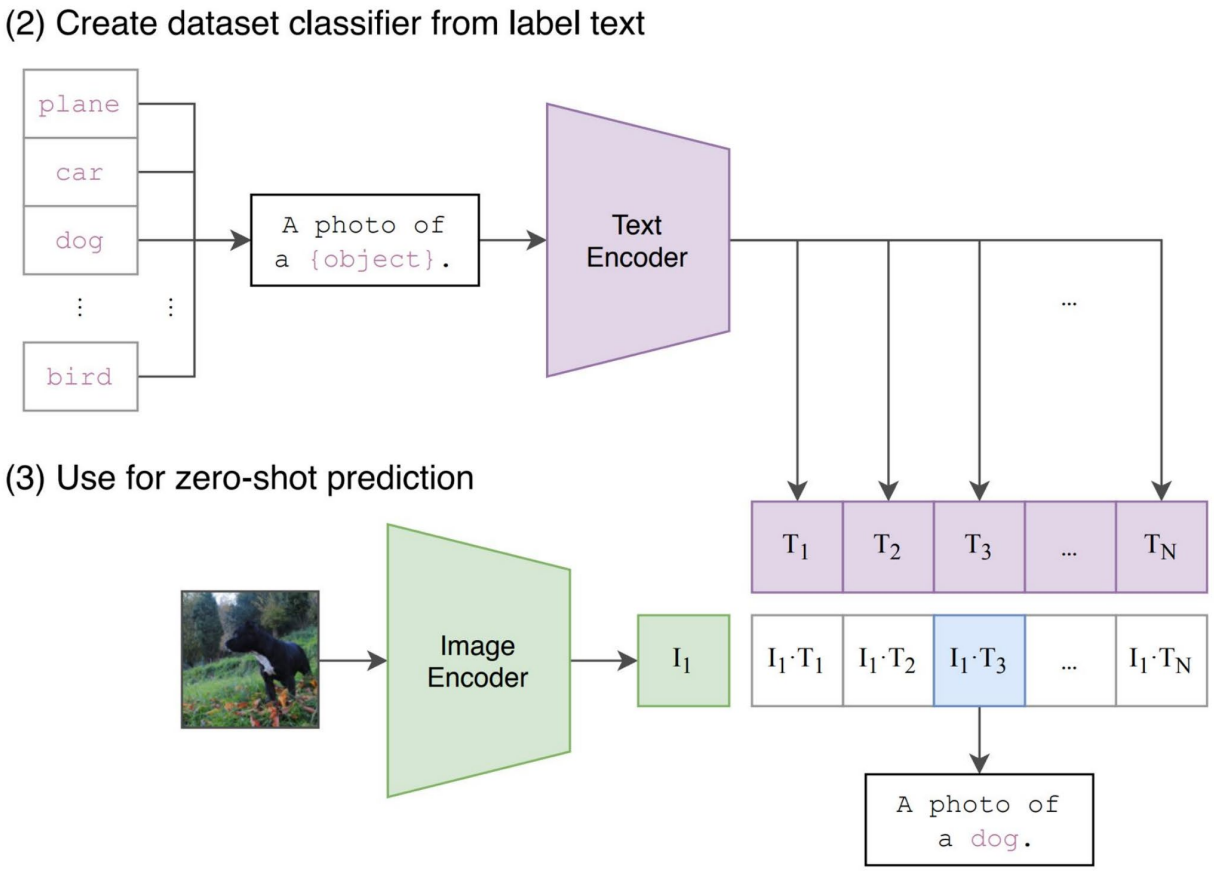


Fig. 1 Summary of CLIP approach [2]

VLA Models

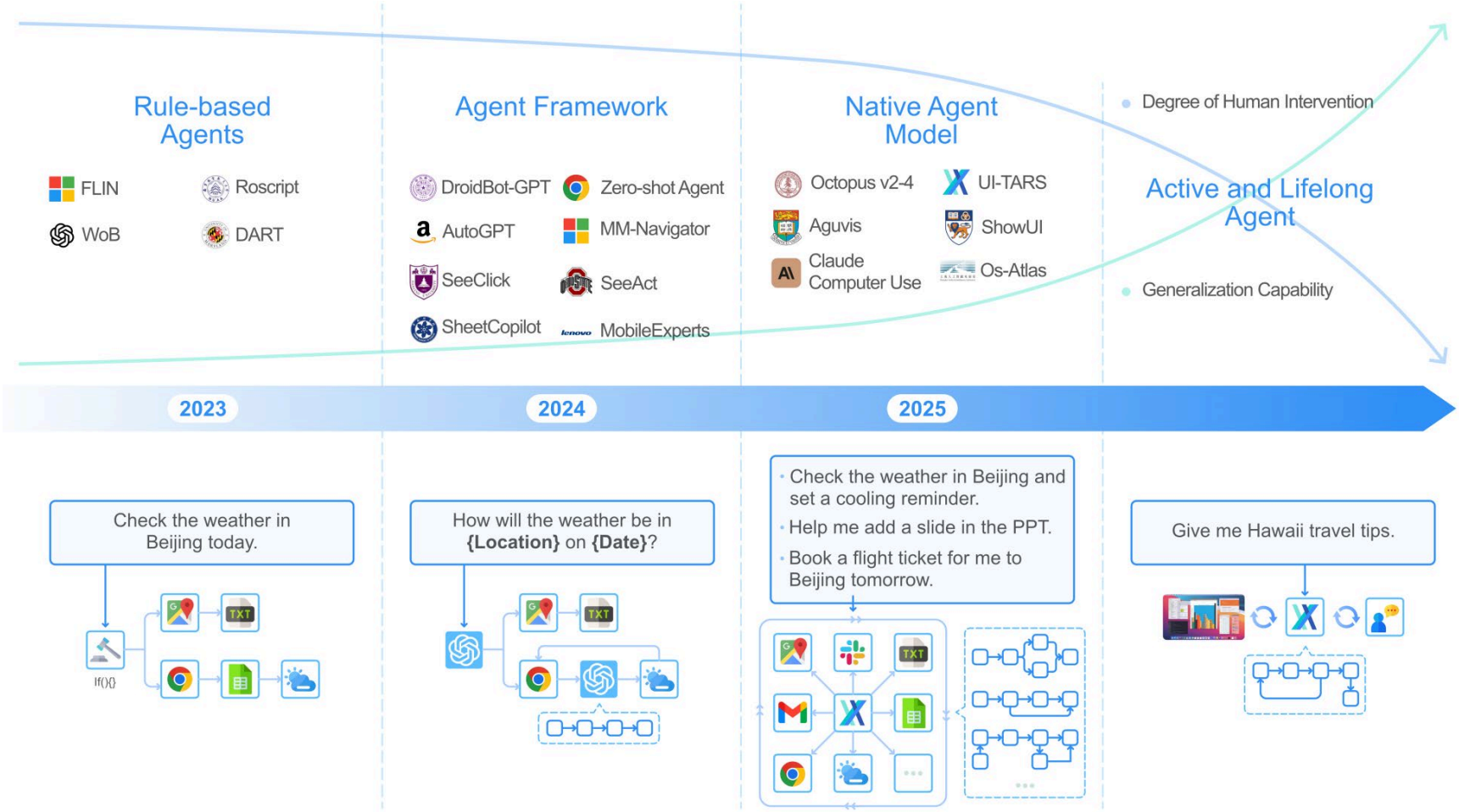


Fig 2. The evolution path for GUI agents [2]

VLA Models

UI-TARS [2]

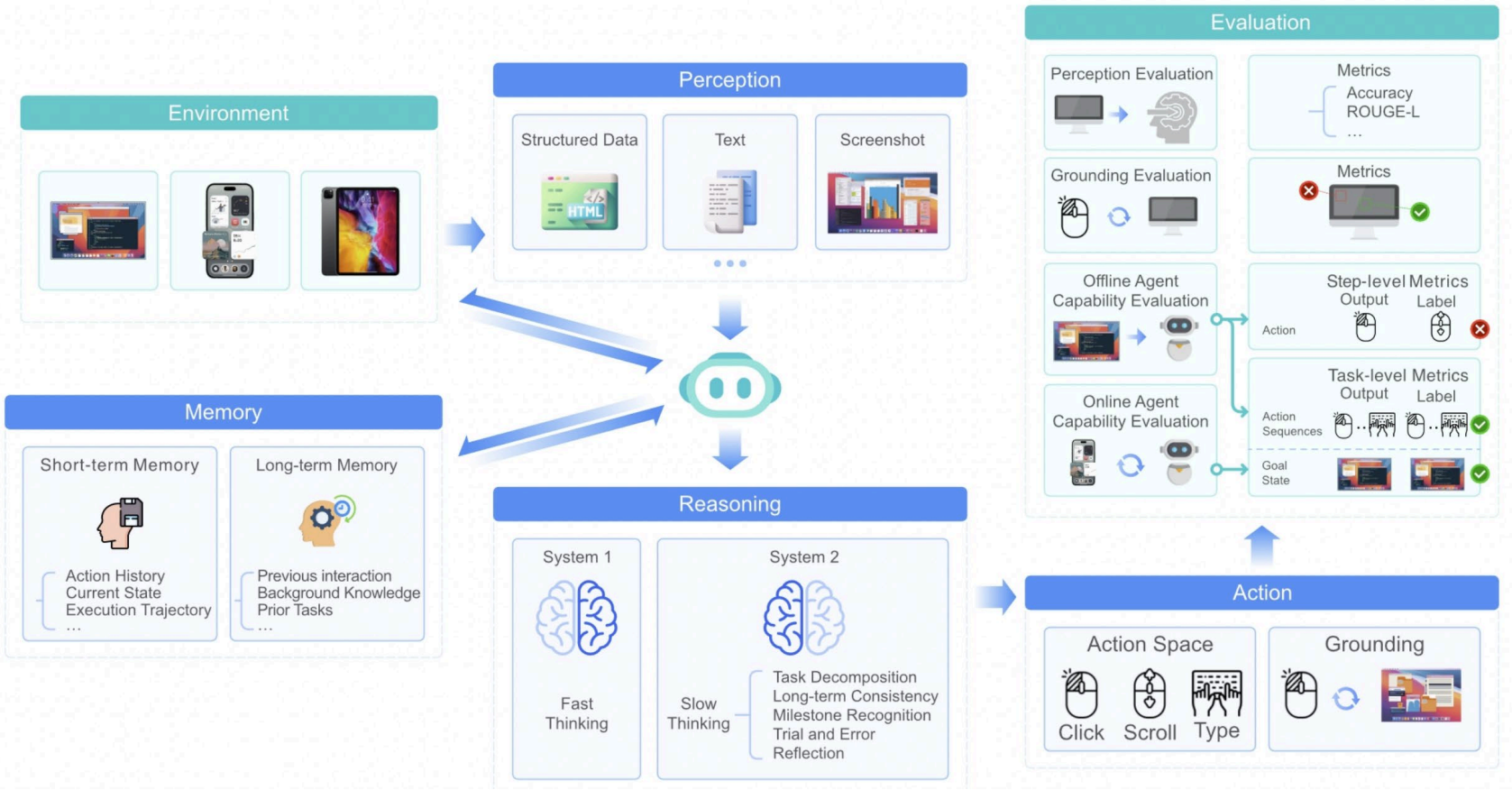


Fig 3. Summary of UI-TARS architecture [2]

VLA Models

ShowUI (2025 CVPR) [3]

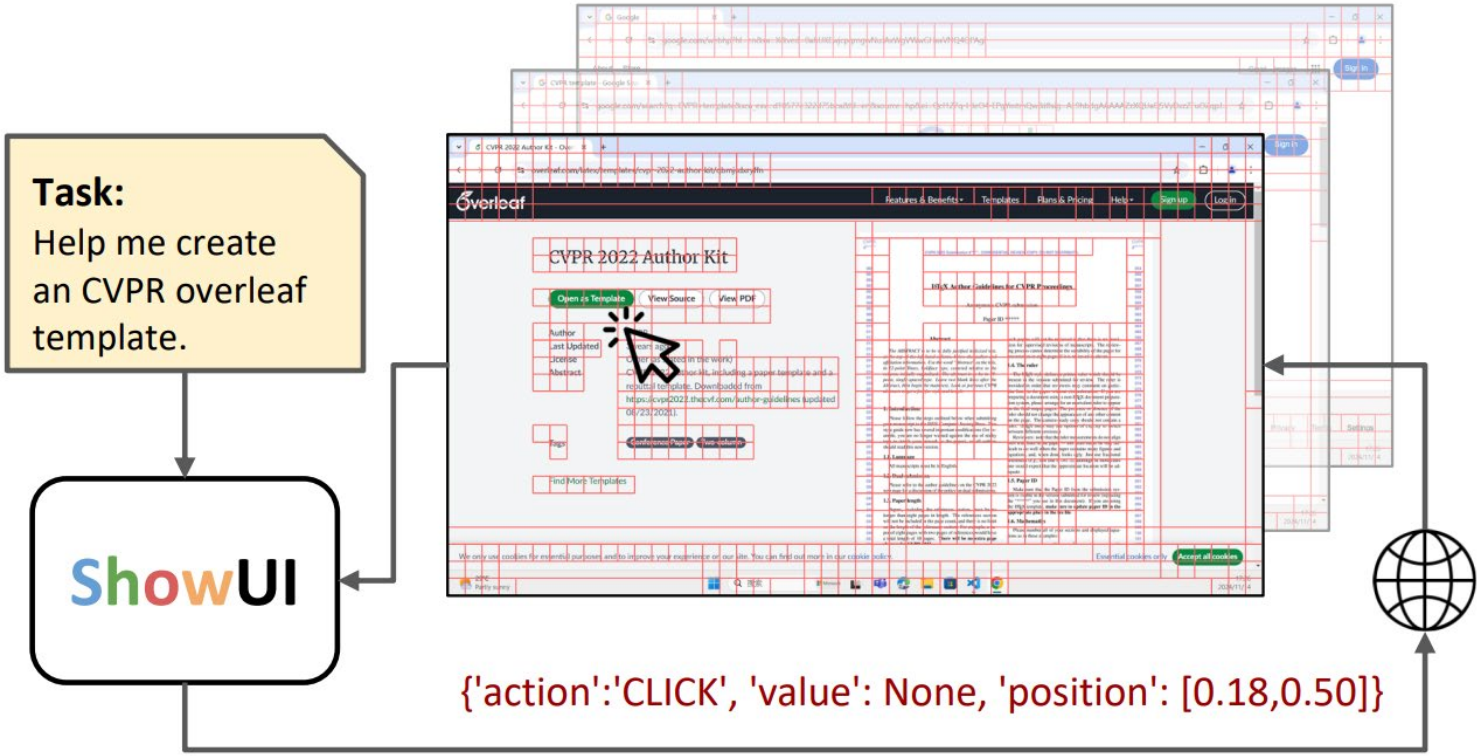


Fig 5. ShowUI is a VLA model for GUI Automation [3]

VLA Models

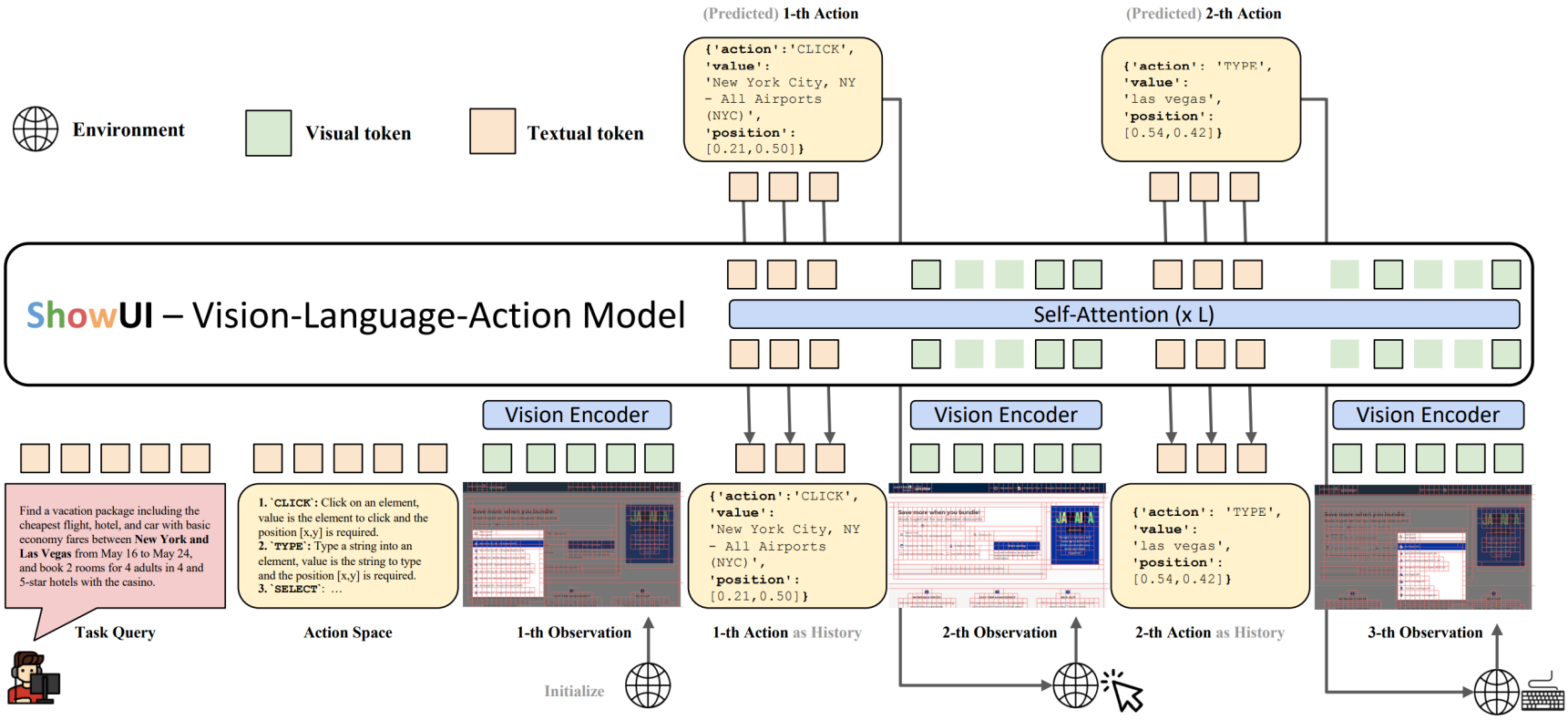


Fig 6. Illustration of ShowUI [3]

Datasets

Dataset	UI Type	Multi Res.	Auto Anno.	#Anno.	Task
Screen 2 Words	Mobile	✗	✗	112k	Screen Summarization
Widget Captioning	Mobile	✗	✗	163k	Element Captioning
RICOSCA	Mobile	✗	✗	295k	Action Grounding
MoTIF	Mobile	✗	✗	6k	Mobile Navigation
RefExp	Mobile	✗	✗	20.8k	Element Grounding
SeeClick Web	Web	✗	✓	271k	Element Grounding
MultiUI	Web, Mobile	✓	✓	3M	Act. & Elem. Ground
UGround-Web	Web	✓	✓	1.3M	Element Grounding
UI REC/REG	Web	✓	✓	400k	Box2DOM, DOM2Box
Ferret-UI	Mobile	✓	✓	250k	Elem. Ground & Ref.
AutoGUI	Web, Mobile	✓	✓	704k	Functionality Ground & Ref.
AccuAnnotate (ours)	**Web, Mobile, Desktop**	✓	✓	**8k**	**Element Ground from GUI Screenshots**

Fig 7. Comparison of datasets (Modified from AutoGUI [4])

Methodology

THE UNIVERSITY OF SYDNEY

AccuAnnotate: Data Collection Pipeline

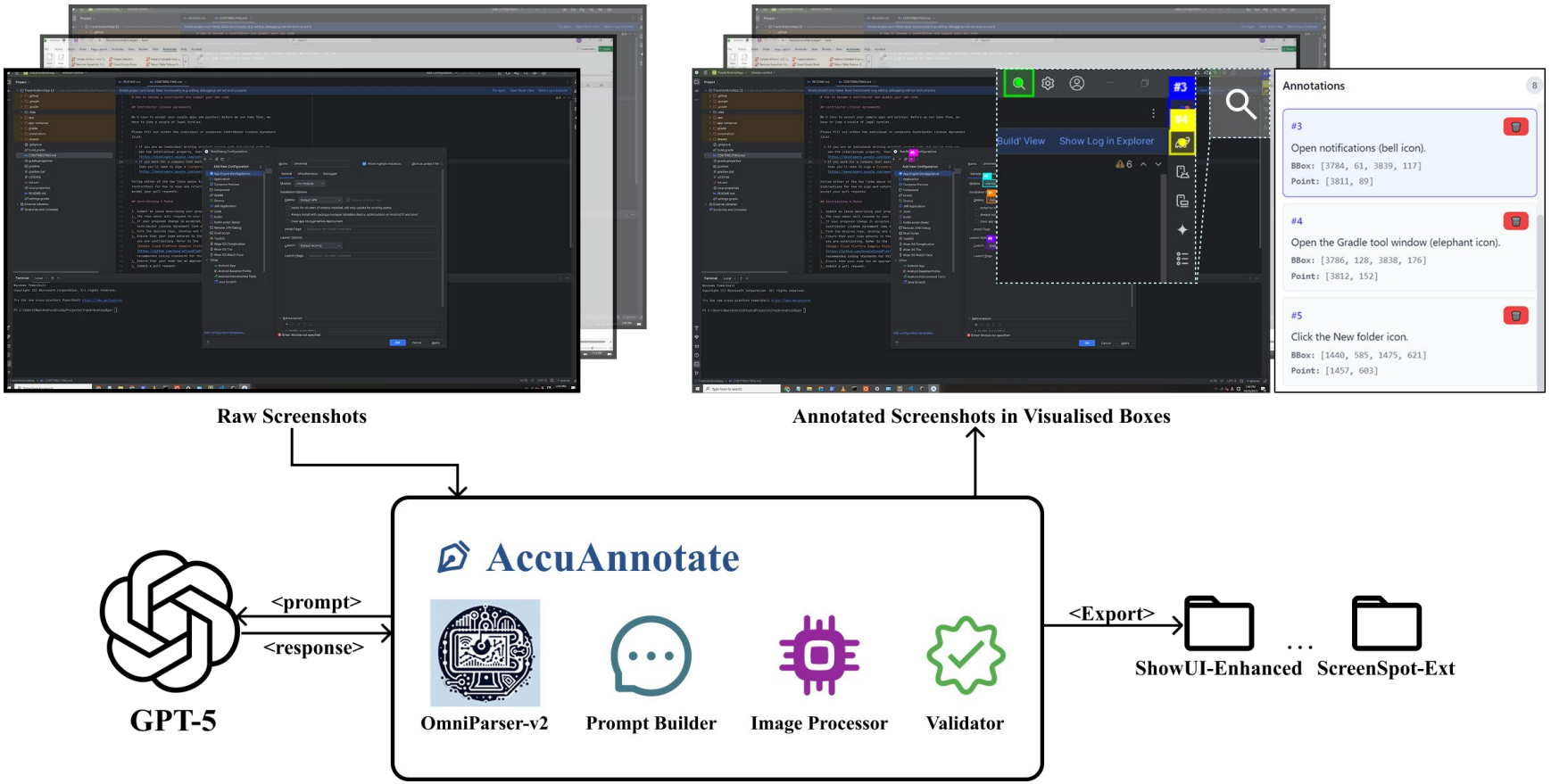


Fig 8. Architecture of AccuAnnotate

AccuAnnotate: Data Collection Pipeline

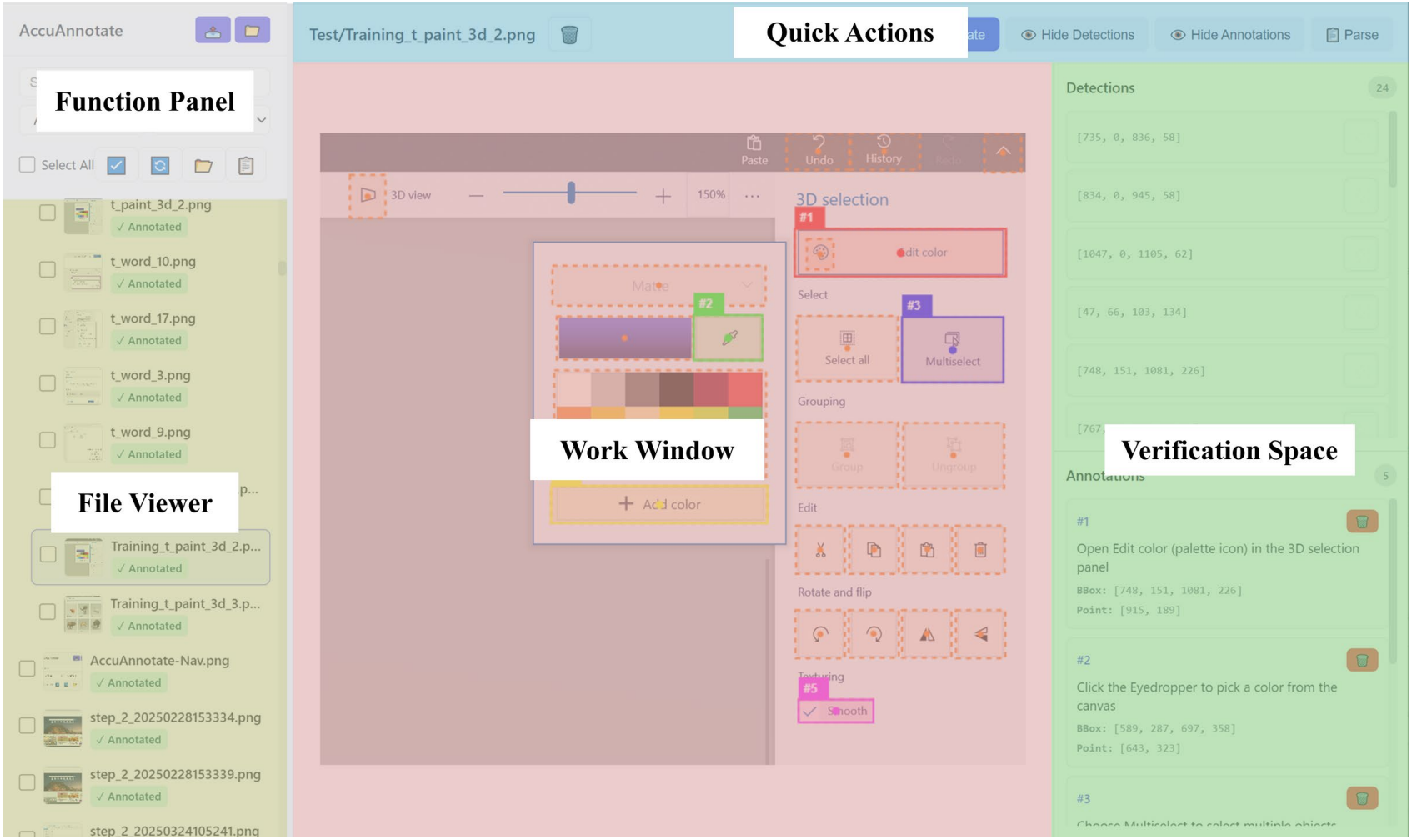


Fig 9. Workspace of AccuAnnotate

AccuAnnotate: Backend API Structure

`GET /api/images`	Paginated DB-backed listing
`GET /api/folders`	Folder tree
`POST /api/upload`	Upload images (size-limited, `secure_filename`)
`GET /api/image/<path>`	Stream image with cache headers
`POST /api/preprocess/<path>`	Return candidate boxes/points (no LLM)
`POST /api/annotate/<path>`	Run full pipeline and persist JSON
`GET /api/annotation/<path>`	Read annotation
`PUT /api/annotation/<path>`	Update annotation
`DELETE /api/annotation/<path>/element/<i>`	Delete element i
`POST /api/batch-annotate`	Launch async job; `GET /status/<job_id>` and SSE `/stream/<job_id>` for progress
`POST /api/export`	Export (ShowUI-Desktop); `GET /api/download-export` for ZIP
`POST /api/deduplicate`	Remove duplicates by base filename

AccuAnnotate-2B: The Compact VLM

- **Input:** (screenshot I, instruction q) → Output: coordinate [x, y].
- **Base Model:** ShowUI-2B (Qwen2-VL style)
- **Method:** Reinforcement Learning

- **Reward Function:**

predicted coordinate $\hat{\mathbf{p}} = [\hat{x}, \hat{y}]$ ground-truth center $\mathbf{p}^\star = [x^\star, y^\star]$

Euclidean distance $d = \|\hat{\mathbf{p}} - \mathbf{p}^\star\|_2$

$$r(\hat{\mathbf{p}}, \mathbf{p}^\star) = \begin{cases} \mathbb{1}(d < \tau) - \alpha_{\text{dist}} \min(d, 0.5) & \text{if } \hat{\mathbf{p}} \text{ is valid,} \\ -1.0 & \text{otherwise (NaN or unparseable).} \end{cases}$$

RL Objective & Stability

- **REINFORCE-style** policy gradient over generated tokens.
- **EMA baseline:**
  - Exponential moving average of reward → reduces variance; ablations show removing EMA is very bad.
- **Entropy regularisation:**
  - Encourages exploration early; annealed down later.
- **Optional KL to reference policy:**
  - Keep model close to the pre-trained policy.
- **Decoding safety:**
  - Greedy warm-up to reduce invalid formats.
  - FSM/regex gate during training so outputs look like [x, y].

Experiments

THE UNIVERSITY OF SYDNEY

Goals

Success Criterion:

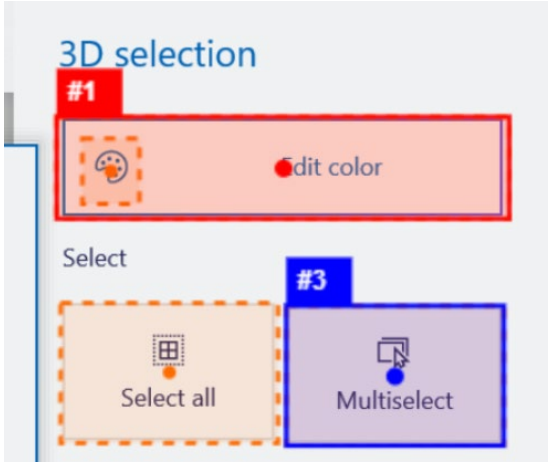
(i) Click-inside-box success ↑

(ii) Point L2 ↓

(iii) Invalid format rate ↓

(iv) Area-Under-Curve (cumulative hit rate) ↑

(v) Distance to box (px) ↓



Training Setup

Platform	Vast.ai
GPU	NVIDIA RTX Pro 6000 WS (96 GB VRAM)
CPU	48 cores
System RAM	255 GB

- **Data:** 400 desktop screenshots, 2,630 tasks
- **Evaluation during training:** small ScreenSpot-desktop subset
- **Metrics:** Success (click-inside-box), L2, DTB, AUC, invalid rate.

Results

Benchmark	Model / Epoch	AUC (%)	DTB (px)	Accuracy (%)
ScreenSpot [2] (334 Tasks)	ShowUI-2B	67.90	78.07	70.06
	ACCUANNOTATE-2B	**71.50**	**70.99**	**72.16**
	Δ	**+5.29%**	**-9.08%**	**+3.00%**
ScreenSpot-Pro [3] (1581 Tasks)	ShowUI-2B	8.88	635.15	13.79
	ACCUANNOTATE-2B	**10.39**	**633.58**	**15.62**
	Δ	**+16.95%**	**-0.25%**	**+13.27%**

TABLE 4.4: ACCUANNOTATE's Performance in Benchmarks

Metrics (quick definitions):

- **DTB:** distance to nearest point on box; 0 if hit.
- **AUC:** cumulative hit-rate over distance thresholds.

Ablation Study

- **Setup:** desktop subset of ScreenSpot (N=334)

Variant	Seeds	Succ (%)↑	L2↓	Invalid (%)↓	Train (min)↓	ΔSucc (pp)	ΔL2	Time×
`full`	2	71.11 ± 1.06	0.102 ± 0.008	0.00	28.0	0.00	0.000	1.00×
`no_fixed_tau`	3	68.56 ± 1.56	0.102 ± 0.004	0.00	60.1	-2.54	-0.000	2.15×
`no_distance`	2	68.71 ± 2.33	0.103 ± 0.009	0.00	28.0	-2.40	0.001	1.00×
`no_ema`	2	63.17 ± 5.08	0.112 ± 0.011	0.00	28.0	-7.93	0.010	1.00×
`no_entropy`	2	66.17 ± 1.27	0.107 ± 0.007	0.00	28.0	-4.94	0.004	1.00×
`no_kl`	2	69.01 ± 1.48	0.103 ± 0.006	0.00	25.9	-2.10	0.000	0.92×
`no_warmup`	2	67.51 ± 1.06	0.103 ± 0.003	0.00	28.0	-3.59	0.001	1.00×
`no_safety`	2	70.36 ± 3.39	0.104 ± 0.001	0.00	28.0	-0.75	0.002	1.00×
`greedy_only`	2	67.51 ± 1.48	0.100 ± 0.007	0.00	27.9	-3.59	-0.002	1.00×

TABLE 4.3: Ablations on the ScreenSpot desktop subset (N=334). Means ± SD across seeds. Δ columns are relative to `full`.

Limitations & Future Work

THE UNIVERSITY OF SYDNEY

Limitations

- **Dependence on OmniParser:** recall/precision cap our labels; misses → missing annotations, false positives → noisy rewards.
- **Dataset bias:** mostly desktop, limited OS/app diversity; limited coverage of mobile/web.
- **Small-scale RL:** few epochs, batch size 1, limited seeds → some seed sensitivity.
- **Evaluation scope:** single-shot clicks on static benchmarks, not full interactive agents.

Future Works

- **Better element discovery:** stronger detectors, ensembles, maybe incorporate accessibility trees.
- **Data diversification:** more apps, platforms, languages; synthetic UIs.
- **Richer rewards & decoding:** size-aware rewards (IoU), constrained decoding with verifier-reranking.
- **Toward full agents:** extend to multistep GUI tasks (OSWorld-style), with retries and planning.

Conclusion

THE UNIVERSITY OF SYDNEY

Answer to the Research Question

- *"How can we automatically produce high-fidelity, pixel-accurate annotations for desktop GUI grounding, and to what extent do these datasets enable reinforcement learning fine-tuning of a compact VLM to improve grounding performance?"*

- "**Yes** – automatically generated, hint-constrained, pixel-level annotations are accurate enough to supervise RL fine-tuning of a compact VLM for desktop GUI grounding."

- "On ScreenSpot benchmarks, this combination improves hit-rates and localisation precision without scaling to massive models."

References

[1] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, G. Krueger, and I. Sutskever, "Learning transferable visual models from natural language supervision," in Proceedings of the 38th International Conference on Machine Learning (ICML), 2021. [Online]. Available: https://proceedings.mlr.press/v139/radford21a.html

[2] Y. Qin, Y. Ye, J. Fang, H. Wang, S. Liang, S. Tian, J. Zhang, J. Li, Y. Li, S. Huang, W. Zhong, K. Li, J. Yang, Y. Miao, W. Lin, L. Liu, X. Jiang, Q. Ma, J. Li, X. Xiao, K. Cai, C. Li, Y. Zheng, C. Jin, C. Li, X. Zhou, M. Wang, H. Chen, Z. Li, H. Yang, H.-Y. Liu, F. Lin, T. Peng, X. Liu, and G. Shi, "UI-TARS: Pioneering automated GUI interaction with native agents," *arXiv preprint* arXiv:2501.12326, 2025. [Online]. Available: https://arxiv.org/abs/2501.12326

[3] K. Q. Lin, L. Li, D. Gao, Z. Yang, S. Wu, Z. Bai, W. Lei, L. Wang, and M. Z. Shou, "Showui: One vision-language-action model for gui visual agent," in CVPR 2025, 2024. [Online]. Available: https://arxiv.org/abs/2411.17465

[4] H. Li, J. Chen, J. Su, Y. Chen, Q. Li, and Z. Zhang, "Autogui: Scaling gui grounding with automatic functionality annotations from llms," arXiv preprint arXiv:2502.01977, 2025, accepted to ACL 2025 Main. [Online]. Available: https://arxiv.org/abs/2502.01977

Q&A

THE UNIVERSITY OF SYDNEY

Thank you

THE UNIVERSITY OF SYDNEY