WHEN MOTION LOGS GO MISSING: SELECTIVE VISUAL RECOVERY FOR PERSISTENT GUI STATE TRACKING

Anonymous authors
Paper under double-blind review

ABSTRACT

Persistent GUI state trackers propagate spatial memory using both rendered screens and recorded actions, implicitly assuming that the action stream is complete. When a motion record is explicitly missing, the tracker applies an incorrect zero update and the resulting coordinate error can persist across subsequent observations. We study this failure mode and formulate recovery as selective substitution: observed actions remain unchanged, while only known motion erasures may be replaced using visual evidence. Our method estimates translation-like motion from consecutive screenshots with linear phase correlation and substitutes the estimate only on missing-action transitions, preserving exact behavior on complete logs. Across five independently trained checkpoints, visual recovery reduces hidden-state and recovery-state error on three frozen stress sets, with all six predeclared comparisons remaining significant after Holm correction. The benefit grows with missing-action severity and remains positive under fixed-budget bursts of one to six consecutive erasures. On held-out GUI-Odyssey trajectories with artificially erased motion logs, linear phase recovery reduces cumulative trajectory-log error from 0.157 to 0.116 and endpoint error from 0.260 to 0.188 at 20% erasure, with positive improvements from 5% through 40% erasure. Under the same trajectory protocol, RAFT-Small provides smaller gains, while a missing-only true-action oracle reveals substantial remaining headroom. On Android-in-the-Wild, linear phase yields the largest gesture-referenced displacement-error reduction among the evaluated estimators. These results show that repairing missing motion inputs can improve persistent GUI state without altering valid action logs. The method targets explicit translation-like motion erasures and does not address arbitrary nonzero action errors or end-to-end GUI-agent control.

1 INTRODUCTION

GUI agents increasingly operate over long interaction trajectories rather than isolated screenshots. Recent work has improved GUI grounding and action prediction across web, mobile, and desktop interfaces (Cheng et al., 2024; Gou et al., 2025; Wu et al., 2025; Lin et al., 2025), while WebArena, OSWorld, and AndroidWorld have shifted evaluation toward multi-step interaction in dynamic environments (Zhou et al., 2024; Xie et al., 2024; Rawles et al., 2025). Long-horizon systems consequently rely on interaction history and temporally accumulated observations (Zheng et al., 2024; Agashe et al., 2025; Chen et al., 2025; Kang et al., 2026), and datasets such as Android-in-the-Wild (AITW) and GUI-Odyssey explicitly record sequences of screenshots and actions (Rawles et al., 2023; Lu et al., 2025). This setting makes spatial consistency across observations increasingly important.

A largely implicit assumption in persistent state propagation is that the recorded motion stream is complete. Consider a target that temporarily leaves the viewport. If a swipe moves the viewport while the target is hidden, its stored coordinate must be propagated by the corresponding motion record. If that record is explicitly missing and represented by a zero-motion sentinel, the tracker instead applies a zero update. The resulting error persists in memory and can remain after the target reappears, as illustrated in Figure 1. We study this *known-erasure* setting rather than arbitrary action

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

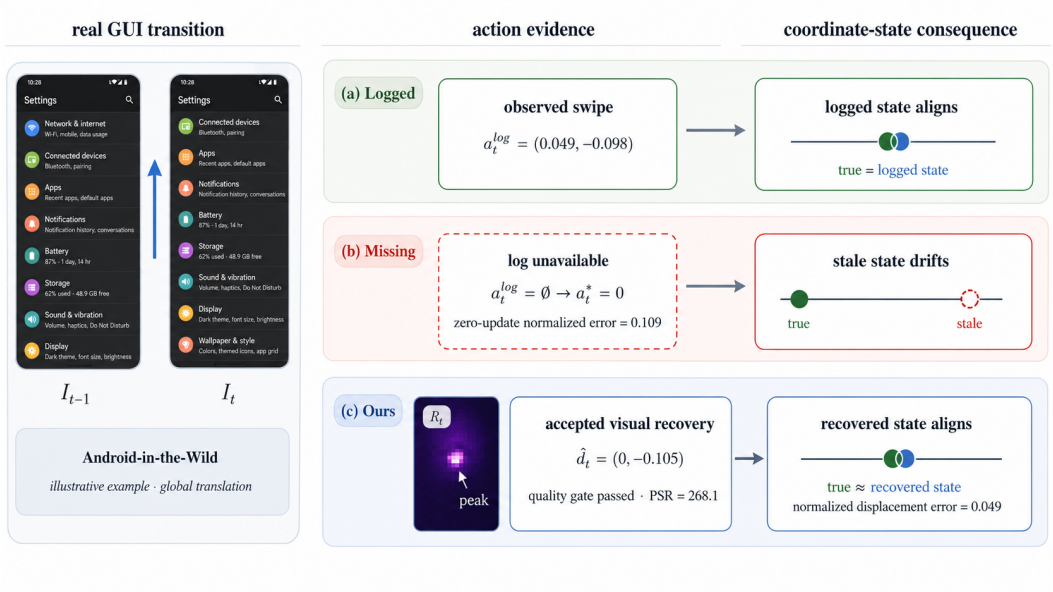

Figure 1: **Missing motion logs cause persistent state drift.** A missing motion record produces a zero update despite viewport movement, causing a persistent coordinate offset. We recover only explicitly erased motion from consecutive screenshots while leaving observed actions unchanged.

corruption: observed nonzero actions are assumed valid and must remain unchanged, while only an explicitly erased motion record may be reconstructed.

We formulate recovery as *selective substitution*. Consecutive screenshots provide a redundant observation of translation-like interface motion, which we estimate using linear phase correlation (Kuglin & Hines, 1975). The estimate is converted to the tracker's action units and substituted only for an explicitly missing motion record before the same unit-consistent coordinate update used for logged actions. A development-frozen reliability screen may reject uncertain visual estimates, but never modifies observed nonzero actions. The resulting method has no additional learned recovery parameters and satisfies *clean-log invariance*: with no missing-action sentinel, its state trajectory is exactly identical to that of the matched coordinate-memory control.

Across five independently trained checkpoints, visual recovery reduces the predeclared recovery-state and hidden-state errors on three frozen stress conditions, with all six primary comparisons remaining significant after Holm correction. Its benefit increases with missing-action severity and persists under fixed-budget bursts of one to six consecutive erasures, while 0% erasure leaves the tracker exactly unchanged. On held-out GUI-Odyssey trajectories, linear phase reduces cumulative trajectory-log error from 0.157 to 0.116 and endpoint error from 0.260 to 0.188 at 20% erasure, with positive improvements from 5% through 40%. Under the same protocol, RAFT-Small yields smaller gains, while a missing-only true-action oracle reveals substantial remaining headroom. On AITW, linear phase also provides the largest gesture-referenced displacement-error reduction among linear phase, corrected tiled phase, RAFT-Small, ORB+RANSAC, and ECC. Representation controls further attribute the state improvement to recovering missing motion rather than to a phase-valued state representation.

Contributions.

- We identify explicitly missing motion logs as a persistent-state failure mode in GUI tracking and formulate recovery as selective substitution under a clean-log invariance requirement.
- We recover translation-like motion from consecutive screenshots only on known erasures and inject it into a unit-consistent coordinate update, while preserving every observed nonzero action and adding no learned recovery parameters.

- We evaluate recovery across five checkpoints using frozen stress conditions, missing-rate and consecutive-erasure tests, held-out GUI-Odyssey trajectories, AITW motion-estimator comparisons, representation controls, and a unit-consistent missing-only oracle.

2 Related Work

GUI grounding and visual action models. Recent GUI agents increasingly improve visual grounding and action prediction from screenshots. SeeClick learns GUI-specific grounding (Cheng et al., 2024), while CogAgent and Ferret-UI target fine-grained interface understanding (Hong et al., 2024; You et al., 2024). UGround, OS-ATLAS, ShowUI, GUI-World, and UI-Ins further advance grounding, high-resolution perception, and GUI-specific action reasoning (Gou et al., 2025; Wu et al., 2025; Lin et al., 2025; Lei et al., 2025; Chen et al., 2026). These methods primarily improve reasoning about currently observed interface content. Our problem instead concerns maintaining spatial state when the target may be hidden and a motion input required for state propagation is explicitly absent.

Temporal context and long-horizon GUI interaction. AITW, WebArena, OSWorld, AndroidWorld, and GUI-Odyssey increasingly evaluate agents over temporally evolving interaction sequences (Rawles et al., 2023; Zhou et al., 2024; Xie et al., 2024; Rawles et al., 2025; Lu et al., 2025). Correspondingly, Synapse uses interaction trajectories as in-context experience (Zheng et al., 2024), Agent S maintains experience for hierarchical computer-use planning (Agashe et al., 2025), GUI-World studies temporal GUI understanding (Chen et al., 2025), and LongHorizonUI uses execution monitoring and compensatory reasoning (Kang et al., 2026). Our focus is complementary: rather than semantic or episodic memory, we study propagation of an explicit coordinate state when a required motion record is known to be missing.

Image registration and visual motion estimation. Phase correlation estimates translation through the normalized cross-power spectrum (Kuglin & Hines, 1975), with localized Fourier upsampling supporting subpixel registration (Guizar-Sicairos et al., 2008). Alternative registration approaches include ORB with robust estimation such as RANSAC (Rublee et al., 2011; Fischler & Bolles, 1981), ECC alignment (Evangelidis & Psarakis, 2008), and learned dense optical flow such as RAFT (Teed & Deng, 2020). Correlation-response statistics including peak-to-sidelobe ratio have also been used to identify unreliable tracking updates (Bolme et al., 2010). Our contribution is not a new registration algorithm. We instead use visual motion as a redundant signal for repairing *known* action erasures: observed actions remain authoritative, visual substitution is restricted to missing records, and complete logs produce exactly the original state trajectory. Reliability screening can reject uncertain substitutions, but our experiments do not assume that screening is uniformly superior to ungated recovery.

3 Method

We study persistent GUI state tracking under explicitly marked motion erasures. Our key principle is *selective substitution*: observed actions remain authoritative, while visual motion may replace only a motion record known to be missing. This restriction separates recovery from general action-error correction and guarantees that complete action logs remain unchanged. Figure 2 summarizes the overall procedure.

3.1 Problem Formulation

Let $I_{1:T}$ denote a sequence of GUI screenshots and

$$a_t = (\mathbf{d}_t, \rho_t, \boldsymbol{\nu}_t, r_t)$$

the recorded transition into frame t, where $\mathbf{d}_t \in \mathbb{R}^2$ is normalized viewport translation, ρ_t is the zoom ratio, $\boldsymbol{\nu}_t \in \mathbb{R}^2$ is the normalized change in viewport extent, and $r_t \in \{0, 1\}$ indicates a reset.

Under the logging protocol considered here, an explicitly missing motion record is represented by a reserved near-zero translation sentinel:

$$m_t = \mathbb{1}\left[\|\mathbf{d}_t\|_2 < \epsilon_{\text{miss}}\right], \qquad \epsilon_{\text{miss}} = 10^{-5}. \tag{1}$$

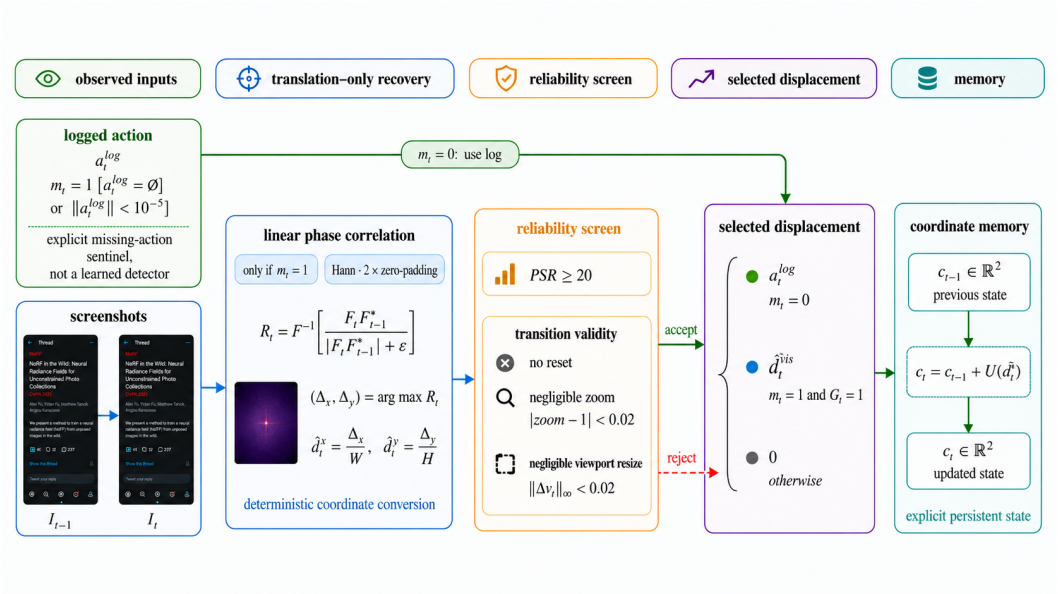

Figure 2: **Selective visual recovery for persistent GUI state tracking.** Observed nonzero actions bypass recovery and remain unchanged. For an explicitly missing motion record, consecutive screenshots provide a translation estimate that may be accepted by a development-frozen reliability screen before the selected displacement updates persistent coordinate memory.

Thus, m_t is part of the observed logging convention rather than the output of a learned fault detector. In particular, the method does not infer whether an arbitrary observed nonzero action is incorrect.

The tracker maintains a persistent target coordinate $\mathbf{p}_t \in \mathbb{R}^2$. Coordinates inside $[0, 1]^2$ correspond to the current viewport, while values outside this range retain the predicted off-screen target position. Let z_t denote cumulative zoom and $\mathbf{v}_t \in \mathbb{R}^2$ the normalized viewport extent. The recovered tracker and its matched control share the same visual frontend, learned parameters, state representation, and propagation rule; they differ only in the motion supplied on explicitly erased transitions.

3.2 Visual Recovery of Missing Motion

When $m_t = 1$, we estimate translation-like motion between consecutive screenshots I_{t-1} and I_t using linear phase correlation (Kuglin & Hines, 1975). Each image is converted to grayscale, mean-centered, and multiplied by a separable Hann window. Let $F_t = \mathcal{F}(G_t)$ denote the resulting two-dimensional Fourier transform. The normalized cross-power spectrum and correlation response are

$$C_t(\boldsymbol{\omega}) = \frac{F_t(\boldsymbol{\omega}) F_{t-1}(\boldsymbol{\omega})^*}{|F_t(\boldsymbol{\omega}) F_{t-1}(\boldsymbol{\omega})^*| + \epsilon}, \qquad R_t = \mathcal{F}^{-1}(C_t). \tag{2}$$

The dominant peak of R_t estimates the translation of rendered image content. In the controlled tracker, the peak is refined by three-point quadratic interpolation along each axis, yielding the normalized screen-space displacement $\mathbf{s}_t = (s_t^x, s_t^y)$. Because rendered content moves opposite to the viewport origin, the corresponding candidate action displacement is

$$\widehat{\mathbf{d}}_t^{\text{vis}} = -\mathbf{s}_t \odot \frac{\mathbf{v}_t}{z_t}. \tag{3}$$

This places the visual estimate in the same action-space convention as the logged translation $\mathbf{d}_t$ before state propagation. For the real-image evaluations, both spatial axes are additionally zero-padded to reduce circular wraparound on larger displacements; dataset-specific preprocessing is given in Appendix E.

3.3 Reliability Screening and Selective Substitution

Visual change does not always correspond to a valid global viewport translation. Local animation, widget motion, or interface replacement may produce a strong but misleading response. We therefore apply a lightweight development-frozen reliability screen before allowing a visual estimate to replace an erased action.

Let r_1 denote the maximum of R_t, and let μ_R and σ_R denote its mean and standard deviation. We define

$$q_t^{\mathrm{psr}} = \frac{r_1 - \mu_R}{\sigma_R + \epsilon}, \qquad q_t^{\mathrm{psr}} \geq 20. \tag{4}$$

Visual substitution is additionally disabled for resets or non-negligible zoom and viewport-resize transitions:

$$b_t = \mathbb{1}[|\rho_t - 1| < 0.02]\,\mathbb{1}[\|\boldsymbol{\nu}_t\|_\infty < 0.02]\,\mathbb{1}[r_t = 0], \qquad g_t = m_t b_t \mathbb{1}[q_t^{\mathrm{psr}} \geq 20]. \tag{5}$$

The displacement supplied to the tracker is then selected as

$$\widetilde{\mathbf{d}}_t = \begin{cases} \widehat{\mathbf{d}}_t^{\mathrm{vis}}, & g_t = 1, \\ \mathbf{d}_t, & g_t = 0. \end{cases} \tag{6}$$

Hence an accepted visual estimate can replace only an explicitly erased action. A rejected erasure retains the recorded zero update, while every observed nonzero action bypasses the recovery route unchanged. The screen is therefore a conservative safeguard rather than a learned error detector or a claimed source of the main performance gain.

3.4 Coordinate Propagation and Clean-Log Invariance

The selected action displacement and persistent coordinates are expressed in different units. We therefore define

$$\boldsymbol{\alpha}_t = \frac{z_t}{z_{t-1}} \odot \frac{\mathbf{v}_{t-1}}{\mathbf{v}_t}, \qquad \mathbf{p}_t = \left(\mathbf{p}_{t-1} - \widetilde{\mathbf{d}}_t \odot \frac{z_t}{\mathbf{v}_t}\right) \odot \boldsymbol{\alpha}_t. \tag{7}$$

For translation-only motion, $\boldsymbol{\alpha}_t = \mathbf{1}$; zoom and viewport-resize transitions use the same recurrence, while a reset replaces the propagated state with the current visual anchor. Recovery therefore changes only the selected motion, not the state-update rule.

If a trajectory contains no missing-action sentinel, then $m_t = 0$ and therefore $g_t = 0$ for every transition. It follows that

$$\widetilde{\mathbf{d}}_t = \mathbf{d}_t \qquad \forall t. \tag{8}$$

Because both trackers subsequently execute the same coordinate recurrence, their complete state trajectories are identical. We refer to this property as *clean-log invariance*. The method targets explicit translation-like motion erasures when consecutive screenshots remain available; it does not detect arbitrary nonzero action errors or model general nonrigid interface motion.

4 Experiments

4.1 Experimental Setup

Controlled evaluation. The controlled benchmark contains 400 sequences split into 280/60/60 train/validation/test sequences. All methods use the same training data, optimization budget, and five independently trained checkpoints. Confirmatory evaluation uses three frozen stress conditions targeting appearance aliases, long-distance target reappearance, and action noise. We report visible-, hidden-, and recovery-state error, where lower is better. Recovery- and hidden-state comparisons across the three conditions form six predeclared primary tests, evaluated using stratified paired bootstrap with Holm correction.

Missing-action evaluation. We erase translation records at rates $\{0, 5, 10, 20, 40\}\%$. A separate frozen set of 80 translation-only sequences evaluates consecutive erasures: exactly 12 translations are removed per sequence and grouped into bursts of length $\{1, 2, 3, 4, 6\}$, holding the total erasure budget fixed.

Table 1: **Persistent-state recovery on frozen stress sets.** Mean error reduction relative to zero update (positive is better), with 95% paired-bootstrap CIs. The six recovery/hidden comparisons remain significant after Holm correction.

Condition	Recovery	Hidden	Visible
Appearance aliases	$+.0294$ $[.0170, .0419]$	$+.0245$ $[.0129, .0364]$	$+.0156$ $[-.0001, .0314]$
Long reappearance	$+.0649$ $[.0488, .0811]$	$+.0590$ $[.0429, .0755]$	$+.0139$ $[-.0027, .0317]$
Action noise	$+.0293$ $[.0159, .0426]$	$+.0206$ $[.0094, .0325]$	$+.0106$ $[-.0017, .0227]$

Table 2: **Consecutive missing-action bursts.** Exactly 12 translations are erased per sequence. Improvements are relative to zero update.

Burst length	Hidden error	Improvement (95% CI)
1	.2669	$+.0488$ $[.0253, .0725]$
2	.2529	$+.0642$ $[.0418, .0875]$
3	.2466	$+.0630$ $[.0390, .0870]$
4	.2614	$+.0500$ $[.0265, .0732]$
6	.2463	$+.0425$ $[.0226, .0618]$

Real GUI evaluation. GUI-Odyssey (Lu et al., 2025) provides trajectory-level evaluation using 51 held-out swipe-bearing episodes drawn from 184 audited trajectories. We erase only motion records while preserving screenshots and unerased actions, and report cumulative and endpoint trajectory-log reconstruction error. AITW (Rawles et al., 2023) provides 303 official-test swipe transitions for gesture-referenced motion-estimator evaluation. These real-data quantities are proxies for logged motion rather than ground-truth world-coordinate state or task success. Full preprocessing, baseline, and statistical details are provided in the appendix.

4.2 Persistent-State Recovery

Table 1 reports the three frozen stress conditions. Visual recovery reduces both predeclared primary errors in every condition, with the largest gains under long-distance reappearance: 0.0649 in recovery-state error and 0.0590 in hidden-state error. All six primary confidence intervals exclude zero and remain significant after Holm correction. Visible-state gains are smaller and not consistently distinguishable from zero, consistent with visible targets being directly re-anchored from the current screenshot while hidden states depend more strongly on accumulated motion.

4.3 Missingness Severity and Consecutive Erasures

At 0% erasure, recovery and its matched control are exactly identical, confirming clean-log invariance. At 5%, the difference is small and its interval includes zero; at 10%, 20%, and 40%, hidden-state error improves by 0.0175, 0.0295, and 0.0512, respectively, while visible-state error improves by 0.0249, 0.0386, and 0.0462.

Table 2 tests temporally concentrated failures while fixing the budget at 12 erased translations. Recovery improves hidden-state error for every burst length from one to six, with all paired-bootstrap intervals excluding zero. The effect is not monotonic with burst length; rather, the fixed-budget experiment shows that recovery remains effective when erasures occur consecutively.

4.4 Recovery on Real GUI Trajectories

Figure 3 evaluates held-out GUI-Odyssey trajectories. At 20% erasure, zero update yields cumulative and endpoint trajectory-log errors of 0.157 and 0.260, whereas ungated linear phase reduces them to 0.116 and 0.188. Its paired-bootstrap improvement intervals exclude zero at every nonzero erasure rate from 5% through 40%, indicating that visual recovery increasingly limits accumulated drift as missing motion grows.

The frozen PSR screen also improves over zero update but is numerically weaker than ungated phase recovery on this protocol, and direct held-out comparisons do not establish a robust screening

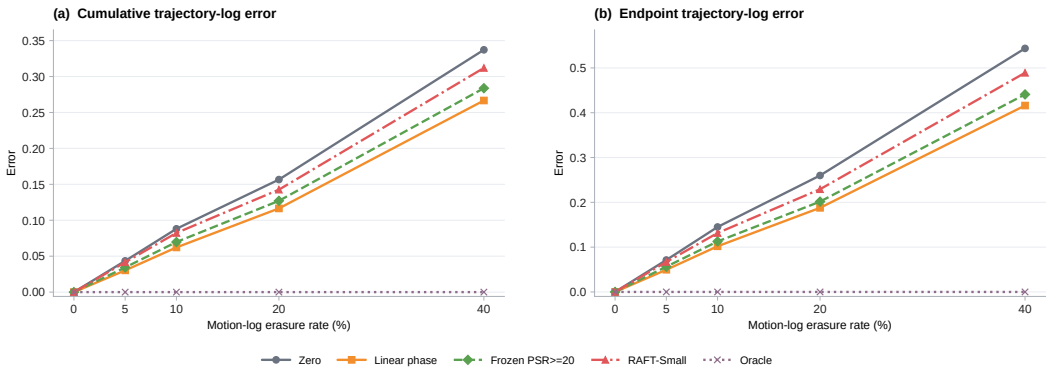

Figure 3: **Held-out GUI-Odyssey trajectory-log reconstruction.** Linear phase reduces cumulative (left) and endpoint (right) error relative to zero update across nonzero erasure rates. RAFT-Small yields smaller gains, while the missing-only true-action oracle shows remaining recoverable headroom. These are gesture-log reconstruction metrics rather than world-coordinate state or task-success measures.

advantage. We therefore treat screening as a conservative safeguard rather than the source of the recovery gain.

RAFT-Small yields smaller trajectory gains than linear phase under the same erased inputs. At 20% erasure, its cumulative and endpoint errors are 0.143 and 0.229, compared with 0.116 and 0.188 for linear phase. This comparison is specific to the translation-dominant recovery setting and is not intended as a general ranking of optical-flow methods.

On AITW, linear phase obtains 78.2% dominant-axis agreement, 71.0% direction agreement, and a gesture-referenced displacement error of 0.331, reducing error relative to zero displacement by 0.130. Corrected tiled phase reduces it by 0.059, while RAFT-Small, ORB+RANSAC, and ECC yield smaller or negative gains under the same evaluation convention.

Finally, the missing-only true-action oracle replaces only erased translations while leaving every nonmissing transition unchanged. At 20% and 40% erasure, visual recovery closes approximately 35.5% and 36.0% of the oracle improvement in hidden-state error. The remaining gap indicates substantial room for better visual displacement recovery. Additional estimator, oracle, representation-control, and reliability analyses are provided in the appendix.

5 Conclusion

We studied a persistent-state failure caused by explicitly missing GUI motion logs. Rather than modifying observed actions, we formulate recovery as selective substitution: consecutive screenshots provide a visual motion estimate only when the corresponding translation record is known to be missing. Observed actions remain authoritative, yielding exact clean-log invariance without additional learned recovery parameters.

Across five independently trained checkpoints, visual recovery reduces recovery-state and hidden-state error under three frozen stress conditions, with larger benefits as missing motion accumulates and persistent improvements under consecutive erasure bursts. On held-out GUI-Odyssey trajectories, linear phase recovery reduces both cumulative and endpoint trajectory-log error from 5% to 40% erasure and provides larger gains than RAFT-Small under the evaluated protocol. AITW experiments further support phase correlation for translation-dominant GUI motion, while a missing-only oracle reveals substantial remaining headroom. These results show that repairing erased motion inputs can prevent a local logging failure from becoming a persistent state error without altering complete action logs.

Limitations. Our method assumes that missing motion records are explicitly identifiable and does not detect or correct arbitrary erroneous nonzero actions. It targets approximately translational view-

port motion and can fail when visual changes are dominated by local, nonrigid, or discontinuous interface dynamics. Moreover, GUI-Odyssey and AITW provide trajectory- and gesture-referenced motion proxies rather than ground-truth world-coordinate state. Our real-data results therefore establish motion-log recovery rather than improved closed-loop GUI-agent task success, which remains an important direction for future work.

AI USE STATEMENT

We used ChatGPT and Codex as research-assistance tools for discussing methodological formulations, checking consistency between manuscript descriptions and implementation artifacts, organizing experimental analyses, and assisting with figure and reference preparation. Generative AI was also used for language editing and improving the clarity and organization of the manuscript. All reported numerical results were obtained from recorded experiments rather than generated or estimated by AI tools. The authors reviewed the AI-assisted material and take full responsibility for the final implementation, analyses, text, figures, and scientific claims.

ETHICS STATEMENT

Our experiments use synthetic interfaces and existing public GUI trajectory datasets. GUI screenshots may contain privacy-sensitive information, and reuse of public data should continue to follow the licenses and access conditions of the original datasets. Persistent GUI state tracking may support useful assistive and automation systems, but could also facilitate unauthorized interface interaction if deployed without appropriate safeguards. Our experiments are entirely offline and evaluate state and motion recovery rather than autonomous execution of real user actions.

REPRODUCIBILITY STATEMENT

Section 3 specifies the missing-action convention, visual displacement estimator, selective-substitution rule, reliability screening, and unit-consistent state update. Section 4.1 describes the controlled and real-data evaluation protocols and statistical analysis. Appendix A provides data splits, action conventions, and implementation details; Appendix B defines the evaluation metrics, matched-comparison protocol, and confirmatory statistical family; Appendix C documents development-only reliability-screen selection; Appendix D reports additional missingness, burst, oracle, and representation controls; and Appendix E details the GUI-Odyssey and AITW evaluation protocols and estimator comparisons. These materials distinguish settings shared across experiments from dataset-specific implementation choices.

REFERENCES

Saaket Agashe, Jiuzhou Han, Shuyu Gan, Jiachen Yang, Ang Li, and Xin Wang. Agent S: An open agentic framework that uses computers like a human. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/394c7c30ea87b5c3521b4d9e9d419071-Abstract-Conference.html.

David S. Bolme, J. Ross Beveridge, Bruce A. Draper, and Yui Man Lui. Visual object tracking using adaptive correlation filters. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2544–2550, 2010. doi: 10.1109/CVPR.2010.5539960.

Dongping Chen, Yue Huang, Siyuan Wu, Jingyu Tang, Liuyi Chen, Yilin Bai, Zhigang He, Chenlong Wang, Huichi Zhou, Yiqiang Li, Tianshuo Zhou, Yue Yu, Chujie Gao, Qihui Zhang, Yi Gui, Zhen Li, Yao Wan, Pan Zhou, Jianfeng Gao, and Lichao Sun. GUI-World: A video benchmark and dataset for multimodal GUI-oriented understanding. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/3926df1a00c9abf056df7bcf253d026a-Abstract-Conference.html.

Liangyu Chen, Hanzhang Zhou, Chenglin Cai, Jianan Zhang, Panrong Tong, Xu Zhang, Quyu Kong, Chen Liu, Yuqi Liu, Wenxuan Wang, Yue Wang, Qin Jin, and Steven

Hoi. UI-Ins: Enhancing GUI grounding with multi-perspective instruction as reasoning. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/hash/5965f3a748a8d41415db2bfa44635cc3-Abstract-Conference.html.

Kanzhi Cheng, Qiushi Sun, Yougang Chu, Fangzhi Xu, Li YanTao, Jianbing Zhang, and Zhiyong Wu. SeeClick: Harnessing GUI grounding for advanced visual GUI agents. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 9313–9332, Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-long.505. URL https://aclanthology.org/2024.acl-long.505/.

Georgios D. Evangelidis and Emmanouil Z. Psarakis. Parametric image alignment using enhanced correlation coefficient maximization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(10):1858–1865, 2008. doi: 10.1109/TPAMI.2008.113.

Martin A. Fischler and Robert C. Bolles. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24 (6):381–395, 1981. doi: 10.1145/358669.358692.

Boyu Gou, Ruohan Wang, Boyuan Zheng, Yanan Xie, Cheng Chang, Yiheng Shu, Huan Sun, and Yu Su. Navigating the digital world as humans do: Universal visual grounding for GUI agents. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=kxnoqaisCT.

Manuel Guizar-Sicairos, Samuel T. Thurman, and James R. Fienup. Efficient subpixel image registration algorithms. *Optics Letters*, 33(2):156–158, 2008. doi: 10.1364/OL.33.000156.

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

Bin Kang, Shaoguo Wen, Yifei Bi, Shunlong Wu, Xinbin Yuan, Rui Shao, Junle Wang, and Zhuotao Tian. LongHorizonUI: A unified framework for robust long-horizon task automation of GUI agent. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://iclr.cc/virtual/2026/poster/10010959.

Charles D. Kuglin and Daniel C. Hines. The phase correlation image alignment method. In *Proceedings of the IEEE International Conference on Cybernetics and Society*, pp. 163–165, 1975.

Weixian Lei, Difei Gao, and Mike Zheng Shou. Grounding multimodal large language model in GUI world. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/31fc85f7461ce71eadf27fb7281973bd-Abstract-Conference.html.

Kevin Qinghong Lin, Linjie Li, Difei Gao, Zhengyuan Yang, Shiwei Wu, Zechen Bai, Stan Weixian Lei, Lijuan Wang, and Mike Zheng Shou. ShowUI: One vision-language-action model for GUI visual agent. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 19498–19508, June 2025. doi: 10.1109/CVPR52734.2025.01816.

Quanfeng Lu, Wenqi Shao, Zitao Liu, Lingxiao Du, Fanqing Meng, Boxuan Li, Botong Chen, Siyuan Huang, Kaipeng Zhang, and Ping Luo. GUIOdyssey: A comprehensive dataset for cross-app GUI navigation on mobile devices. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 22404–22414, October 2025. doi: 10.1109/ICCV51701.2025.02080.

Christopher Rawles, Alice Li, Daniel Rodriguez, Oriana Riva, and Timothy Lillicrap. AndroidInTheWild: A large-scale dataset for android device control. In *Advances in Neural Information Processing Systems*, volume 36, 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/bbbb6308b402fe909c39dd29950c32e0-Abstract-Datasets_and_Benchmarks.html.

Christopher Rawles, Sarah Clinckemaillie, Yifan Chang, Jonathan Waltz, Gabrielle Lau, Marybeth Fair, Alice Li, William E Bishop, Wei Li, Folawiyo Campbell-Ajala, Daniel Kenji Toyama, Robert James Berry, Divya Tyamagundlu, Timothy P Lillicrap, and Oriana Riva. AndroidWorld: A dynamic benchmarking environment for autonomous agents. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=il5yUQsrjC.

Ethan Rublee, Vincent Rabaud, Kurt Konolige, and Gary Bradski. ORB: An efficient alternative to SIFT or SURF. In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 2564–2571, 2011. doi: 10.1109/ICCV.2011.6126544.

Zachary Teed and Jia Deng. RAFT: Recurrent all-pairs field transforms for optical flow. In *Computer Vision–ECCV 2020*, volume 12347 of *Lecture Notes in Computer Science*, pp. 402–419. Springer, 2020. doi: 10.1007/978-3-030-58536-5_24.

Zhiyong Wu, Zhenyu Wu, Fangzhi Xu, Yian Wang, Qiushi Sun, Chengyou Jia, Kanzhi Cheng, Zichen Ding, Liheng Chen, Paul Pu Liang, and Yu Qiao. OS-ATLAS: Foundation action model for generalist GUI agents. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/0faa4bc5f522076947a030273629d4fe-Abstract-Conference.html.

Tianbao Xie, Danyang Zhang, Jixuan Chen, Xiaochuan Li, Siheng Zhao, Ruisheng Cao, Toh Jing Hua, Zhoujun Cheng, Dongchan Shin, Fangyu Lei, Yitao Liu, Yiheng Xu, Shuyan Zhou, Silvio Savarese, Caiming Xiong, Victor Zhong, and Tao Yu. OSWorld: Benchmarking multimodal agents for open-ended tasks in real computer environments. In *Advances in Neural Information Processing Systems*, volume 37, 2024. doi: 10.52202/079017-1650.

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. In *Computer Vision–ECCV 2024*, volume 15122 of *Lecture Notes in Computer Science*, pp. 240–255. Springer, 2024. doi: 10.1007/978-3-031-73039-9_14.

Longtao Zheng, Rundong Wang, Xinrun Wang, and Bo An. Synapse: Trajectory-as-exemplar prompting with memory for computer control. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/hash/52f050499cf82fa8efb588e263f6f3a7-Abstract-Conference.html.

Shuyan Zhou, Frank F. Xu, Hao Zhu, Xuhui Zhou, Robert Lo, Abishek Sridhar, Xianyi Cheng, Tianyue Ou, Yonatan Bisk, Daniel Fried, Uri Alon, and Graham Neubig. WebArena: A realistic web environment for building autonomous agents. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=oKn9c6ytLx.

A Implementation and Training Details

Controlled tracking data. The controlled benchmark contains 400 sequences divided into 280 training, 60 validation, and 60 test sequences. All matched model comparisons use the same data partitions and optimization budget. We train five independently initialized checkpoints using seeds 11, 22, 33, 44, and 55. Model selection is performed independently for each seed using the corresponding validation performance, and all reported comparisons use matched checkpoints and evaluation sequences.

Action and coordinate conventions. Translation actions are represented in normalized viewport coordinates. In the controlled benchmark, the horizontal and vertical action components are normalized by the nominal viewport width and height, respectively, i.e., 800 pixels horizontally and 600 pixels vertically. Let $\mathbf{v}_t$ denote the viewport size and z_t the current scale. The scale- and viewport-change factor is

$$\boldsymbol{\alpha}_t = \frac{z_t}{z_{t-1}} \odot \frac{\mathbf{v}_{t-1}}{\mathbf{v}_t}, \tag{9}$$

and persistent coordinates are propagated as

$$\mathbf{p}_t = \left(\mathbf{p}_{t-1} - \widetilde{\mathbf{d}}_t \odot \frac{z_t}{\mathbf{v}_t} \right) \odot \boldsymbol{\alpha}_t. \tag{10}$$

The zero-update control, visual-recovery condition, and missing-only oracle all use the same state recurrence. They differ only in the displacement supplied on transitions whose motion record is explicitly erased.

Missing-action convention. The experiments consider a known-erasure setting in which a missing motion record is explicitly represented by the reserved zero-motion sentinel used by the tracker. Visual recovery is considered only on such transitions. Logged nonzero actions remain authoritative and are never replaced by visually estimated alternatives. Consequently, when no missing-action sentinel occurs, the recovery condition follows exactly the same state-update trajectory as its matched coordinate-memory control.

Visual displacement recovery. For an erased motion transition, consecutive screenshots are used to estimate translation-like image displacement with linear phase correlation. The estimated displacement is converted to the action convention of the corresponding benchmark before being supplied to the persistent coordinate update in Equation 10. The same recovery principle is used for the controlled benchmark, GUI-Odyssey, and AITW, while coordinate conversion and preprocessing follow the conventions of each dataset. We therefore do not assume that a single pixel-space registration configuration is numerically interchangeable across all three data sources.

B Evaluation Metrics and Statistical Protocol

Persistent-state errors. The controlled benchmark distinguishes states according to target visibility and temporal context. Visible-state error measures localization error when the target is directly observable in the current viewport. Hidden-state error measures persistent coordinate error while the target is outside the current viewport. Recovery-state error measures the state after a previously hidden target becomes observable again. Lower values indicate more accurate persistent state.

Matched comparisons. Controlled comparisons are paired at the checkpoint and sequence levels. The same trained checkpoint, sequence realization, missing-action mask, and frozen evaluation condition are used for both members of each comparison. We report mean paired differences together with 95% confidence intervals obtained by stratified paired bootstrap.

Confirmatory family. The predeclared confirmatory family consists of recovery-state and hidden-state comparisons under the appearance-alias, long-reappearance, and action-noise stress conditions. These six primary tests are corrected jointly using the Holm procedure. The missing-rate, consecutive-erasure, estimator, oracle, and reliability-screen analyses provide additional characterization and are not included in this confirmatory family.

Real-GUI metrics. GUI-Odyssey does not provide ground-truth world coordinates for the persistent state considered in the controlled benchmark. We therefore reconstruct the cumulative trajectory implied by the available gesture log and evaluate cumulative trajectory-log error, endpoint trajectory-log error, and missing-action displacement error. AITW similarly compares visual displacement estimates with displacement derived from recorded gestures. These quantities are trajectory- or gesture-referenced proxies rather than direct measurements of true viewport motion, persistent world-coordinate state, or end-to-end task success.

C Development-Frozen Reliability Screening

The reliability-screen configuration is selected exclusively on the development cohort `fusion_dev5200/dropout20`, which contains 50 sequences generated with seed 5200. This development cohort is disjoint from the held-out stress and missing-rate cohorts used for final evaluation. The same five trained checkpoints corresponding to seeds 11, 22, 33, 44, and 55 are used in the matched development comparison.

Table 3: **Development-only reliability-screen selection.** Lower error is better. The final configuration is selected without using any held-out evaluation result.

Configuration	Development error
No screening	.2738
PSR only	**.2633**
PSR + peak gap	.2659
PSR + peak gap + coherence	.2727
Full screening rule	.2712

Before development evaluation, we fix five candidate configurations: no screening, PSR-only screening, PSR+peak-gap screening, PSR+peak-gap+coherence screening, and the original full screening rule. The associated thresholds are specified before evaluating the development cohort. The selection criterion favors the least complex screened candidate whose mean gain over ungated recovery is nonnegative and whose development error lies within one paired-bootstrap standard error of the best development result.

PSR-only screening achieves the lowest development error and is therefore frozen with $\tau_{\mathrm{PSR}} = 20$. Peak-gap, coherence, and maximum-displacement thresholds are disabled in this selected configuration. The resulting rule is frozen before subsequent held-out evaluation.

Importantly, the held-out experiments do not establish a consistent advantage of PSR screening over ungated visual recovery. The sequence-weighted pooled PSR-only improvement over ungated recovery is $+0.0075$, with a 95% paired-bootstrap confidence interval of $[-0.0007, 0.0221]$. The corresponding held-out differences for the appearance-alias, long-reappearance, and action-noise conditions are $+0.0375$, $+0.0057$, and $+0.0022$, respectively, with all three confidence intervals including zero. Missing-rate comparisons between PSR-only and ungated recovery likewise remain close to zero. We therefore interpret reliability screening as an optional conservative rejection mechanism rather than an independently established source of the main recovery gain.

D Additional Controlled Experiments

D.1 Missing-Action Severity

We additionally vary the fraction of erased translation records while keeping the remaining sequence content unchanged. At 0% erasure, visual recovery and the matched control are exactly identical, empirically confirming clean-log invariance. At 5% erasure, the paired confidence interval includes zero. The recovery advantage becomes clearer as more motion information is removed.

At 10%, 20%, and 40% erasure, visible-state error is reduced by 0.0249, 0.0386, and 0.0462, respectively. Hidden-state error is reduced by 0.0175, 0.0295, and 0.0512, respectively. Thus, the benefit becomes more pronounced at higher erasure rates, particularly for hidden state whose coordinate must be maintained without direct visual re-anchoring.

D.2 Consecutive Missing-Action Bursts

Independent random erasures do not capture failures in which several consecutive motion records disappear. We therefore construct an additional frozen translate-only benchmark containing 80 sequences. Exactly 12 translations are erased from each sequence, while the same total erasure budget is grouped into bursts of length 1, 2, 3, 4, or 6. This protocol changes the temporal concentration of missing actions without changing their total number.

The improvement remains positive for every tested grouping, and all reported paired-bootstrap intervals exclude zero. The effect is not monotonic in burst length. Because the number of erased actions is fixed, longer bursts simultaneously reduce the number of distinct missing segments; the experiment is therefore intended to test persistence under contiguous erasures rather than establish a monotonic burst-length scaling law.

Table 4: **Controlled consecutive-erasure evaluation.** Exactly 12 translations are erased per sequence. Hidden error is reported for the recovered tracker; improvement is relative to zero update.

Burst length	Hidden error	Improvement (95% CI)
1	.2669	$+.0488$ $[.0253, .0725]$
2	.2529	$+.0642$ $[.0418, .0875]$
3	.2466	$+.0630$ $[.0390, .0870]$
4	.2614	$+.0500$ $[.0265, .0732]$
6	.2463	$+.0425$ $[.0226, .0618]$

We also evaluate error eight transitions after the beginning of each erased segment. The corresponding improvements are $+0.0469$, $+0.0617$, $+0.0626$, $+0.0531$, and $+0.0500$ for burst lengths 1, 2, 3, 4, and 6, respectively. These results further indicate that the benefit persists beyond the immediately erased transition.

D.3 Unit-Consistent Missing-Only Oracle

To estimate recoverable headroom, we construct an oracle that replaces only translations labeled as erased. Every observed nonmissing action remains unchanged. For an erased transition, the recorded horizontal and vertical gesture components are normalized by their corresponding nominal viewport dimensions, i.e., 800 pixels horizontally and 600 pixels vertically, before being supplied to the same state recurrence used by the other conditions.

This missing-only construction is important. Replacing every transition with recorded actions would conflate recovery of erased inputs with correction of errors or noise already present in observed actions and would therefore not isolate the failure mode studied in this paper. The oracle is also not expected to produce zero total state error, because visual-frontend error and errors associated with nonmissing transitions remain.

At 20% erasure, visual recovery improves hidden-state error by 0.02949, whereas the missing-only oracle improves it by 0.08315. Visual recovery therefore closes approximately 35.5% of the oracle improvement. At 40% erasure, the corresponding improvements are 0.05115 and 0.14206, respectively, corresponding to approximately 36.0% of the recoverable oracle gap. The remaining difference indicates substantial room for improving visual displacement estimation without changing the selective-substitution formulation.

D.4 Representation Controls

We test whether the observed improvement could instead arise from representing persistent state in a phase-valued space. In one control, the target state is represented on the unit circle while using the same recovered displacement as the continuous coordinate-memory tracker. The resulting predictions agree with continuous coordinate memory to approximately 10^{-8} numerical precision. Changing the state parameterization alone therefore does not explain the recovery gain.

A second control accumulates phase-spectrum state directly instead of converting the estimated displacement into the coordinate recurrence. This alternative increases error by 0.0503 relative to the corresponding coordinate-memory recovery condition. Together, these controls support the interpretation that the improvement comes from recovering the missing motion input rather than from storing persistent state in the phase domain.

E Real-GUI Evaluation Protocol and Results

E.1 GUI-Odyssey Trajectory Reconstruction

We audit 184 GUI-Odyssey trajectories, of which 97 contain at least one linked swipe, yielding 277 linked swipe transitions. The final held-out evaluation uses 51 swipe-bearing episodes selected according to the frozen split. Development and held-out trajectories are disjoint.

Table 5: **Held-out GUI-Odyssey trajectory reconstruction.** Each entry reports cumulative / endpoint trajectory-log error. Lower is better. All methods receive the same screenshots and erasure masks.

Erasure	Zero	PSR20	Linear phase	RAFT-Small
0%	.000/.000	.000/.000	.000/.000	.000/.000
5%	.043/.071	.034/.056	**.030/.050**	.042/.067
10%	.088/.145	.070/.113	**.062/.102**	.082/.132
20%	.157/.260	.127/.202	**.116/.188**	.143/.229
40%	.337/.544	.284/.441	**.267/.416**	.312/.489

For each missing-rate condition, the original screenshots and all unerased actions are preserved. Only selected motion records are replaced by the missing-action sentinel. The erasure masks are nested across rates, so that a higher erasure rate extends rather than independently resamples the lower-rate mask. This construction reduces unnecessary variation when comparing recovery behavior across missingness levels.

GUI-Odyssey gestures use a 1000×1000 normalized action convention. Image-space displacement estimates are therefore converted to full-screen fractions before trajectory accumulation. For each trajectory, we reconstruct the cumulative displacement implied by the remaining and recovered gesture records. We report cumulative trajectory-log error, endpoint trajectory-log error, and displacement error restricted to erased actions. These metrics evaluate reconstruction of the recorded gesture trajectory rather than ground-truth world-coordinate state or end-to-end task success.

At 5% erasure, ungated linear phase reduces cumulative trajectory-log error by 0.0132 with a 95% paired-bootstrap confidence interval of $[0.0018, 0.0295]$, and endpoint error by 0.0217 $[0.0034, 0.0482]$. At 10%, the corresponding improvements are 0.0259 $[0.0082, 0.0515]$ and 0.0431 $[0.0126, 0.0832]$. At 20%, cumulative and endpoint improvements reach 0.0401 $[0.0190, 0.0689]$ and 0.0722 $[0.0344, 0.1210]$. At 40%, they increase to 0.0705 $[0.0377, 0.1102]$ and 0.1276 $[0.0660, 0.1997]$. Thus, the paired intervals for ungated phase recovery exclude zero at every nonzero erasure rate in this held-out trajectory evaluation.

PSR-only recovery also improves over zero update, but is numerically weaker than ungated linear phase on this protocol. At 20% erasure, for example, PSR-only recovery obtains cumulative and endpoint errors of 0.1270 and 0.2016, compared with 0.1165 and 0.1878 for ungated phase. This observation is consistent with the held-out controlled diagnostics and further motivates treating reliability screening as optional rather than as the main source of recovery performance.

RAFT-Small produces smaller gains than linear phase under the same erased inputs. At 20% erasure, RAFT-Small reduces cumulative error by 0.0140 with a confidence interval of $[-0.0084, 0.0338]$, while its endpoint improvement is 0.0307 $[0.0010, 0.0613]$. We do not claim a statistically significant pairwise advantage of phase over RAFT because a separate phase-versus-RAFT paired confidence interval is not used for this comparison. Instead, these results show that the simpler translation-oriented estimator produces larger gains under the evaluated trajectory-reconstruction protocol.

The missing-only true-action oracle obtains zero trajectory reconstruction error because this particular GUI-Odyssey evaluation reconstructs the same gesture log from which the erased actions are drawn. This property is specific to the trajectory-log metric and should not be confused with the controlled persistent-state oracle, where visual-frontend and nonmissing-transition errors remain.

E.2 Additional GUI-Odyssey Burst Analysis

As a secondary real-data analysis, we erase a single contiguous swipe burst from each eligible held-out GUI-Odyssey episode. Because longer bursts require episodes containing sufficiently long consecutive swipe runs, the cohort size decreases rapidly with burst length.

The first three burst lengths have intervals excluding zero, whereas the length-four interval crosses zero. Because only ten eligible episodes remain at length four, we do not interpret this secondary analysis as evidence of monotonic improvement with burst length.

Table 6: **Secondary GUI-Odyssey burst analysis.** Improvement is measured relative to zero update. Cohort size decreases with burst length and the analysis is therefore treated as descriptive.

Burst	Episodes	Mean error	Improvement (95% CI)
1	51	.2201	$+.0536\ [.0265, .0814]$
2	23	.5715	$+.0961\ [.0418, .1519]$
3	12	.9798	$+.1320\ [.0251, .2538]$
4	10	1.1638	$+.1463\ [-.0113, .3122]$

Table 7: **Motion estimation on 303 AITW swipe transitions.** Axis and direction measure agreement with the recorded gesture. L2 error is gesture-referenced displacement error, and gain denotes its reduction relative to the zero-displacement baseline.

Estimator	Axis (%)	Direction (%)	L2 error	Gain
Linear phase	**78.2**	71.0	**.331**	**+.130**
Corrected tiled phase	71.0	59.4	.402	+.059
RAFT-Small	56.4	**71.6**	–	+.023
ORB+RANSAC	61.4	53.1	–	-.345
ECC	42.9	38.0	–	+.005

E.3 AITW Motion-Estimator Evaluation

AITW evaluation uses the same 303 swipe transitions from the official test split for every estimator. Because AITW records gestures rather than ground-truth viewport displacement, each visual estimate is compared with the displacement implied by the corresponding gesture. The resulting measurements are therefore gesture-referenced proxies.

For tiled phase estimation, tile-normalized displacement is converted back to the coordinate system of the evaluated crop by multiplying the horizontal and vertical components by the corresponding tile-width/crop-width and tile-height/crop-height ratios. This corrected conversion is used for all reported tiled-phase results.

Linear phase obtains 78.2% dominant-axis agreement and 71.0% direction agreement, with gesture-referenced L2 error 0.331. Relative to the zero-displacement baseline, this corresponds to an error reduction of 0.130. Corrected tiled phase produces a smaller gain of 0.0586. RAFT-Small yields a gain of 0.023, whose paired confidence interval includes zero, while ORB+RANSAC and ECC do not provide comparable improvements under this protocol. Among the evaluated estimators, linear phase therefore yields the largest reduction in gesture-referenced displacement error.

These results should not be interpreted as a general benchmark of optical-flow or registration algorithms. The comparison addresses a narrower question: which estimator provides useful translation-like displacement for recovering missing GUI motion under the evaluated screenshot and gesture conventions.

E.4 RAFT-Small Runtime

For the 277 GUI-Odyssey screenshot pairs used in the trajectory experiment, the measured RAFT-Small forward-pass time is 12.856 seconds in total, or approximately 46.4 ms per pair. This timing covers model inference and excludes parquet loading, report generation, and other evaluation bookkeeping. Phase-correlation and RAFT measurements were not collected under an identical end-to-end timing pipeline, so we do not use these values to make a hardware-matched efficiency claim.

F Additional Diagnostics and Limitations

Held-out screening diagnostics. The development-selected PSR screen does not show a robust held-out advantage over ungated recovery. On the appearance-alias condition, the PSR-only difference relative to ungated recovery is $+0.0375$ with a 95% interval of $[-0.0064, 0.1198]$. The

corresponding differences are $+0.0057$ $[-0.0022, 0.0155]$ for long reappearance and $+0.0022$ $[-0.0024, 0.0071]$ for action noise. Under 5%, 10%, 20%, and 40% random erasure, the PSR-only differences relative to ungated recovery are -0.0008, approximately zero, $+0.0010$, and $+0.0003$, respectively, with all intervals crossing zero. These results reinforce the interpretation that the recovery signal itself, rather than the reliability screen, accounts for the main improvement.

Threshold diagnostics. Additional frozen threshold analyses show that more restrictive screening is not uniformly beneficial. In particular, configurations that impose the original coherence threshold can perform worse than less restrictive alternatives on some held-out conditions. These diagnostics were conducted after the development configuration had been frozen and were not used to retune the final threshold. We therefore report them only as evidence that visual-recovery reliability is condition dependent.

Real-GUI reliability annotation. We prepared a blinded manual-annotation package containing 250 AITW candidate transitions selected deterministically from the same 303-swipe cohort. Before/after screenshots are separated from the corresponding screen diagnostics so that reliability labels can be assigned without observing the screen decision. Human labels were not available for the present evaluation. Accordingly, we do not report real-GUI true-positive rate, false-acceptance rate, precision, recall, or balanced accuracy for the reliability screen.

Scope of the failure model. The experiments address explicit erasure of translation-like motion records. They do not evaluate arbitrary erroneous nonzero actions, semantic action misclassification, adversarial logging errors, or general nonrigid screen transformations. Because observed nonzero actions remain authoritative, the method intentionally cannot repair an incorrect action that is present in the log but not marked as missing.

Scope of the real-data evaluation. The GUI-Odyssey experiments reconstruct accumulated gesture logs rather than world-coordinate target states, and the AITW analysis uses recorded gesture displacement as a proxy for viewport motion. Neither experiment measures closed-loop GUI-agent task completion. Establishing whether improved motion-log recovery translates into higher long-horizon task success therefore requires an end-to-end evaluation beyond the scope of the present study.