COUNTERFACTUAL EVIDENCE-AWARE CREDIT WITH MATCHED CONTROLS FOR SINGLE-STEP REINFORCEMENT LEARNING OF GUI VISUAL GROUNDING

Anonymous authors*
Paper under double-blind review

ABSTRACT

A GUI grounding policy can click the right location for the wrong visual reason. A binary click reward cannot tell whether the policy relies on the referred interface element or on a shortcut, and editing only the target cannot tell either, because a click can move after any image edit. We introduce **counterfactual evidence-aware credit with matched controls** (CEAC), which applies the same edit to the target and to same-size control regions outside it and turns the difference in the policy's response into training credit. We measure the matched gap: how much more the click moves when the target is edited than when the controls are edited. Compared with five other training objectives, CEAC increases the matched gap on held-out mobile screens by 15% to 22%, with no statistically significant accuracy loss. On PC/Web, CEAC also has the largest gap, and its lead over three of the five other objectives is significant after multiple-comparison correction. Mask and blur edits give the same gain, which comes from a larger response to target edits while control responses stay flat. Matched controls can thus train a policy to rely on evidence at the target location, which the task reward alone does not require.

1 INTRODUCTION

Visual grounding turns a screenshot and an instruction into a screen coordinate to click. It is the step at which the agent's reading of the screen becomes an action (Cheng et al., 2024; Hong et al., 2024; You et al., 2024). Reinforcement learning with verifiable rewards usually judges that action with a single bit. A click inside the target box succeeds, and any other click fails (Bai et al., 2024; Xia & Luo, 2025). This reward is cheap to compute and matches the task, but it says nothing about why a correct click was correct. In Figure 1, a policy can click "7" by recognising its label or by recalling its usual top-left position. Both clicks receive the same reward, so the reward cannot separate recognition of the target from a layout shortcut.

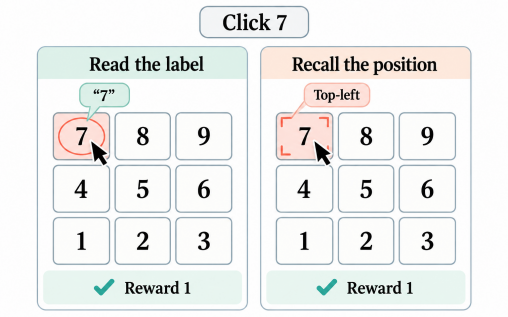


Figure 1: Reading the label "7" or recalling its usual position earns the same reward, so success alone does not show which cue was used.

A natural test is to edit the target, for example by masking it with a blank patch or blurring it, and see how far the click moves; we call this distance the *target response*. On its own, the target response is ambiguous. A mask or blur changes local texture, contrast, and image statistics, so even an edit to an irrelevant region can move the click. We therefore apply the same edit to *control regions*, which lie elsewhere on the screen and have the target's width and height, and call the resulting click movement the *control response*. The *matched gap* is the target response minus the average control response. A positive gap means that the click reacts more to an edit of the referred element than to equally large edits elsewhere. Accuracy tells us *whether* the click is correct; the matched gap tells us whether it

*Code: https://anonymous.4open.science/r/CEAC-013F/

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.

depends *specifically on the target location*, which neither accuracy nor the target response alone can show.

We introduce **counterfactual evidence-aware credit with matched controls** (CEAC), which uses this comparison during policy optimisation. For each sampled action, CEAC computes its *token support*, the mean log probability of the action's tokens, with the target masked and with each control masked. If masking the target lowers the support more than masking the controls, the action receives extra *evidence credit*, weighted by how close the click lands to the target. The task reward and the policy update stay unchanged. Evaluation applies the same target-versus-control idea to click movement instead of token support, with both mask and blur edits. Because the training signal and the evaluation measure differ, whether the first raises the second is an empirical question.

We compare CEAC with five comparator objectives (ICA, SRPO, VPPO, ViGoRL, and strict GRPO, which uses only the task reward) under the same training setup, with five seeds and complete held-out PC/Web and mobile test sets. On mobile, CEAC keeps comparable accuracy and increases the matched gap by 15.3% to 21.7% over every comparator; all five paired differences remain significant after Holm correction. On PC/Web, the observed gain is 7.1% to 12.6%, and three of the five differences remain significant. Splitting the gap into target and control responses shows that the gain comes from the target response. Because mask and blur edits agree, the gain is also unlikely to be an artefact of mask-only evaluation.

Our contributions are:

- the matched gap, a measure that separates the policy's response to the target from its general sensitivity to local image edits;
- an evidence credit rule that adds this contrast, measured on token support, to the task advantage while keeping the task objective, optimiser, and policy update unchanged; and
- a five-seed study in which CEAC increases the mobile matched gap by 15.3% to 21.7% over five comparators, with separate analyses of accuracy, PC/Web results, and optimisation.

2 Related Work

GUI grounding and visual agents. GUI agents operate across web, mobile, and desktop environments such as Mind2Web, AndroidInTheWild, and OSWorld (Deng et al., 2023; Rawles et al., 2023; Xie et al., 2024). Grounding systems improve coordinate prediction with vision-language models specialised for interfaces and with screen parsing (Cheng et al., 2024; Hong et al., 2024; You et al., 2024; Lu et al., 2024), and recent work scales these ideas to more applications and screen resolutions (Li et al., 2025a; Gou et al., 2025; Wu et al., 2025; Lin et al., 2025; Li et al., 2025b). These methods mainly evaluate whether the predicted coordinate is correct. We study a different problem, in which a click is correct but depends only weakly, or not specifically, on the referred element.

Reinforcement learning for GUI control. Policy gradient learning ranges from REINFORCE to clipped policy updates (Williams, 1992; Schulman et al., 2017). GRPO replaces a learned critic with a baseline computed from a group of sampled actions (Shao et al., 2024). Verifiable and environment rewards have since become common in reasoning and device control systems (Guo et al., 2025; Bai et al., 2024; Xia & Luo, 2025). Recent objectives add signals for progress, image dependence, or a second look at part of the image (Fei et al., 2026; Huang et al., 2026; Sarch et al., 2025). GUI-G1 adds a reward for box size and adjusts the GRPO objective for response length and sample difficulty (Zhou et al., 2025). SE-GUI combines a dense distance-based reward with attention-guided selection of training examples (Yuan et al., 2025). These methods go beyond binary success, but they do not compare the same action under an edit of the target and under matched edits elsewhere on the screen. CEAC uses this comparison to assign evidence credit.

Counterfactual credit and perturbation evidence. Difference rewards and counterfactual baselines measure an agent's contribution by comparing it with a controlled alternative (Wolpert & Tumer, 2002; Foerster et al., 2018). Visual explanation methods probe predictions with perturbations, masks, or counterfactual images (Fong & Vedaldi, 2017; Petsiuk et al., 2018; Goyal et al., 2019), and faithfulness studies show that perturbation maps can look plausible and still be misleading (Adebayo et al., 2018; Hooker et al., 2019; Yeh et al., 2019; Jacovi & Goldberg, 2020). CEAC

applies the idea of a controlled alternative to action credit. The target and control edits have the same type, size, and shape, so subtracting the control response removes the general effect of editing the image, which the target response alone would include.

3 Problem Setup

Let $x = (I, c)$ consist of a screenshot I and an instruction c. The policy $\pi_\theta(a \mid x)$ outputs an action, a sequence of tokens that decodes to a click position $\hat{p} \in \mathbb{R}^2$. With ground-truth target box $B^\star(x)$, the task reward is

$$r_i = \mathbb{1}[\hat{p}_i \in B^\star(x)]. \tag{1}$$

Each episode has a single step, so the problem is a contextual bandit. GRPO samples K actions from the behaviour policy and forms

$$A_i^{\text{task}} = \mathrm{Z}(r_i). \tag{2}$$

Here Z subtracts the group mean and divides by the population standard deviation. If that deviation is below 10^{-8}, it returns zero for every action. This advantage ranks sampled actions by task success alone. Every click inside the box receives the same reward, whatever image region the policy relied on. CEAC adds credit to individual actions without changing the task reward or the policy update.

4 Counterfactual Evidence-Aware Credit

CEAC adds evidence credit to the task advantage defined above. Figure 2 shows its four stages: score the same sampled actions again with different regions masked, compare their support under control and target masks, weight these contrasts by click quality, and add the resulting credit to the task advantage. All stages use the same group of sampled actions.

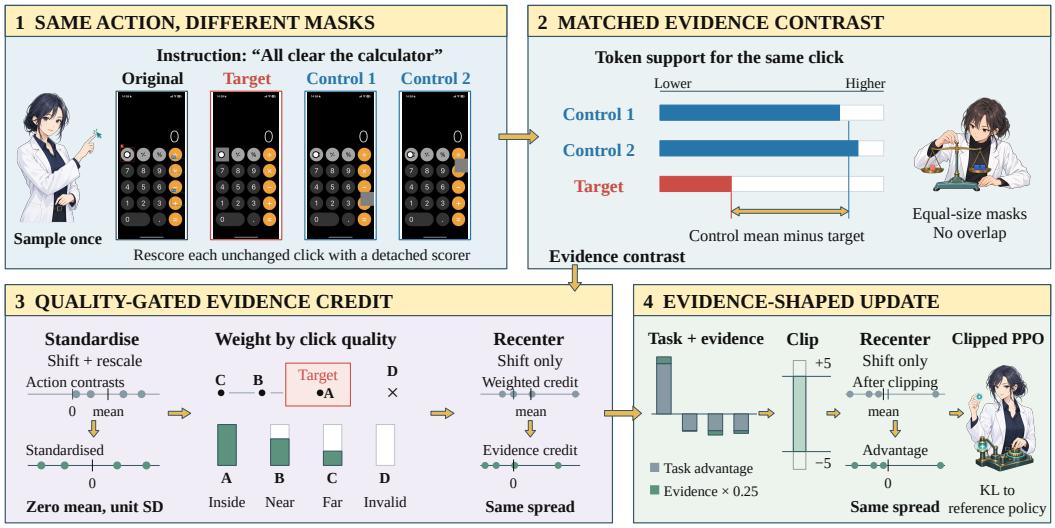


Figure 2: CEAC training pipeline. (1) Sample actions on the original screenshot, then score each unchanged action again with the target masked and with each of two matched controls masked. (2) Compare token support: mean score with a control masked minus score with the target masked. (3) Standardise the contrasts within the action group, weight them by click quality, and recentre. (4) Add evidence credit to the task advantage, clip, recentre, and update with clipped PPO and a KL penalty to the reference policy. Bars and points only illustrate the operations.

4.1 Scoring the same action under matched masks

The training signal should reward support that depends on the target and ignore support that any local edit would remove. In stage 1, actions are sampled on the original screenshot. Each action is then kept fixed and scored again on masked copies of the screenshot. For a sampled action a_i,

let ℓ_i^t be its token support, the mean log probability of its tokens, after masking the target region $R_\star = B^\star(x)$. A deterministic rule selects two control regions R_{c_1}, R_{c_2} with the target's width and height, which overlap neither the target nor each other. If ℓ_{ij}^c is the token support after masking control j, stage 2 forms the raw evidence contrast

$$e_i = \frac{1}{2}\sum_{j=1}^{2} \ell_{ij}^c - \ell_i^t. \tag{3}$$

A positive e_i means that masking the target removes more support for a_i than masking a control does. This contrast is the training counterpart of the matched gap used for evaluation (Section 5.1). Both subtract a control response, but e_i measures token support, whereas the gap measures how far the click moves. All log probabilities come from a detached copy of the behaviour policy, through which no gradients flow, so the contrast does not add a second gradient path.

4.2 Weighting credit by click quality

A large contrast alone is not enough, because a click that depends strongly on the target can still land in the wrong place. Stage 3 therefore standardises the contrast within the sampled group and then multiplies it by a click-quality score $q_i \in [0, 1]$ that falls with distance from the target:

$$E_i = \mathrm{Z}(e_i), \qquad U_i = \mathrm{Center}(q_i E_i). \tag{4}$$

The inside, near, far, and invalid examples in panel 3 of the figure illustrate this weighting. Correct clicks have $q_i = 1$. For an incorrect but parseable click, $q_i = \exp[-d_i/(0.15\ \mathrm{diag}(I))]$, where d_i is the shortest Euclidean distance to the target box and $\mathrm{diag}(I)$ is the image diagonal. An invalid output receives zero. We call this weighting the *quality gate*. It stops the evidence term from rewarding a strong response when the click lands far from the target.

4.3 Policy update with evidence credit

Stage 4 adds the evidence credit to the task advantage:

$$\begin{gathered}\widetilde{A}_i = \mathrm{Center}\big[\mathrm{Clip}\big(A_i^{\mathrm{task}} + \lambda U_i, -c_{\mathrm{adv}}, c_{\mathrm{adv}}\big)\big], \\ \lambda = 0.25, \qquad c_{\mathrm{adv}} = 5.\end{gathered} \tag{5}$$

The two terms are added, the sum is clipped, and the result is centred again, in that order. The resulting $\widetilde{A}_i$ enters the same clipped PPO objective and KL penalty as every comparator. The reference policy stays fixed during training. CEAC changes only how credit is divided among the sampled actions; the optimiser is unchanged. The task reward and the quality gate still depend on where the click lands relative to the target. The evidence term does not penalise the use of this position information; it only adds a comparison between the target and control regions.

Algorithm 1 keeps sampling separate from scoring. Each action is sampled once from the original screenshot. The target and control masks change only the image used to score that same action. A contrast between different sampled actions would mix a change in the image with a change in the action itself. Scoring the same action again isolates the effect of each masked region on its support.

Algorithm 1 CEAC advantage construction

Require: behaviour policy, group size K
$\lambda = 0.25$, clip $c_{\mathrm{adv}} = 5$
for each state $x = (I, c)$ **do**
  sample $\{a_i\}_{i=1}^K$; compute A_i^{task}
  construct $R_\star$ and matched R_{c_1}, R_{c_2}
  for $i = 1, \ldots, K$ **do**
    score $\ell_i^t, \ell_{i1}^c, \ell_{i2}^c$ under masks
    $e_i \leftarrow \frac{1}{2}\sum_j \ell_{ij}^c - \ell_i^t$
  end for
  standardise all e_i to obtain E_i
  $U_i \leftarrow \mathrm{Center}(q_i E_i)$
  form $\widetilde{A}_i$ by Eq. (5)
  update with clipped PPO and the KL penalty
end for

In Figure 2, standardisation subtracts the group mean and divides by the standard deviation, whereas both Recenter steps only subtract the group mean: panel 3 centres the quality-weighted evidence, and panel 4 the combined advantage after clipping. Because the scorer is detached, these steps only produce the advantage for the policy update and add no separate training objective. The task reward still decides which clicks succeed, and the evidence term only separates actions by what their support depends on.

5 Evaluation and Experimental Setup

At evaluation, we ask whether the predicted click responds specifically to the target. This measurement is separate from the token-support contrast used in training. We first define it and then describe the data, edits, and training settings shared by all methods.

5.1 Matched gap for evaluation

Evaluation applies the same measurement to every method. With m matched controls, the matched gap is the target response minus the mean control response:

$$\mathrm{FG} = \Delta_{\mathrm{target}} - \frac{1}{m}\sum_{j=1}^{m} \Delta_{\mathrm{control},j}. \tag{6}$$

Specifically, let $\hat{p}(I)$ be the greedy (most likely) click. We call each evaluation edit a *probe*, and let $O_R(I)$ apply probe $O \in \{M, B\}$, either a mask or a local Gaussian blur, to region R. The response is the distance the click moves, divided by the image diagonal:

$$\Delta_R^O = \frac{\|\hat{p}(I) - \hat{p}(O_R(I))\|_2}{\mathrm{diag}(I)}. \tag{7}$$

For each probe, the evaluator subtracts the mean response over four controls, which are the same regions for every method:

$$\mathrm{FG}^O = \Delta_\star^O - \frac{1}{4}\sum_{j=1}^{4} \Delta_{c_j}^O, \qquad \mathrm{FG} = \tfrac{1}{2}(\mathrm{FG}^M + \mathrm{FG}^B). \tag{8}$$

A larger positive FG means stronger *target specificity*: editing the referred element moves the click more than editing control regions of the same size. The gap does not measure task accuracy or overall usefulness, and should be read together with accuracy and with its target and control responses.

5.2 Shared experimental setup

Data and splits. Using ScreenSpot (Cheng et al., 2024), we train on 510 PC/Web examples and evaluate on two held-out domains: 130 PC/Web examples (63 desktop PC and 67 web) and 500 mobile examples. The mobile split changes interface density, aspect ratio, and interaction conventions, but the task is still to click the referred element. A separate 128-example validation split is used only for development analyses. It is not used to choose seeds, checkpoints, or reported test results for the *confirmation runs*, the final five-seed runs whose results we report. Every confirmation run is evaluated on the complete test sets, with no examples dropped or hand-picked.

Matched controls. Controls have the target's width and height and overlap neither the target nor each other. Candidate positions lie on a fixed 5×5 grid. They are ranked by how similar they look to the target, with an extra term that favours controls whose centres are spread apart. CEAC uses the first two valid controls during training. Evaluation uses the first four and applies the same control regions, mask probe, and blur probe to every objective. Two of the 512 training examples and two of the 502 mobile examples (0.39% and 0.40%, respectively) had no room for four valid controls. We removed them before the confirmation runs. The validation and PC/Web test sets were unchanged, and every objective uses the same resulting splits.

Training and comparators. We fine-tune Qwen2.5-VL-3B-Instruct (Bai et al., 2025) with LoRA (Hu et al., 2022) (rank 16, scale 32, dropout 0.05) for three epochs with each of five seeds, $\{101, 211, 307, 401, 503\}$. The learning rate is 10^{-5}. Each run uses group size four, two PPO passes, clip $\epsilon = 0.2$, and reference KL coefficient 0.04. The six objectives are CEAC, ICA, SRPO, VPPO, ViGoRL, and strict GRPO. All six share the backbone, training examples, optimiser, decoding, PPO update, and evaluator, and differ only in the credit assigned to sampled actions. ICA and VPPO add credit for the drop in an action's log probability under a central mask or a global blur, respectively. SRPO adds credit for a valid coordinate and format, and ViGoRL adds a bonus when the click stays correct after cropping. CEAC uses mask credit from two controls with $\lambda = 0.25$. Exact objective definitions are in the appendix.

Statistical analysis. We report mean±sample standard deviation (SD) over the five seeds, which are the same for every objective. For each domain and metric, CEAC is paired with every comparator by seed and example. Confidence intervals use 10,000 hierarchical bootstrap resamples, first over seeds and then over screenshot groups within a seed. Holm correction adjusts for the five comparator tests (Efron & Tibshirani, 1993; Holm, 1979). All 30 runs use the intended training configuration, and every reported table row is computed from them.

Evaluation coverage. Each of the 30 runs is evaluated on all 630 test examples. For every example, the evaluator records the unedited prediction and the responses to edits of the target and four controls under two probes, giving 18,900 run–example evaluations and 189,000 measurements on edited images. These measurements make estimates within a run more precise, but statistical tests treat the seed as the unit of replication. No checkpoints, examples, or qualitative cases are hand-picked.

6 Results and Analysis

6.1 Accuracy and target specificity

We first compare accuracy and matched gap. We then test whether CEAC's larger gap is reliable and where it comes from, and finally examine how the credit rule changes training.

Table 1: Confirmation results from 30 runs (six objectives, five seeds). Gap is the matched gap, target response minus control response, averaged over mask and blur; it measures target specificity, not accuracy. Values are mean±sample SD.

Method	PC/Web accuracy	PC/Web gap	Mobile accuracy	Mobile gap
CEAC	0.775±0.011	**0.203±0.012**	0.879±0.006	**0.302±0.009**
ICA	0.752±0.017	0.180±0.009	0.870±0.005	0.259±0.014
SRPO	0.772±0.004	0.190±0.005	0.881±0.006	0.255±0.012
VPPO	0.703±0.048	0.185±0.012	0.843±0.018	0.258±0.012
ViGoRL	0.777±0.012	0.189±0.004	0.882±0.007	0.262±0.014
GRPO	0.785±0.015	0.186±0.003	0.884±0.006	0.248±0.010

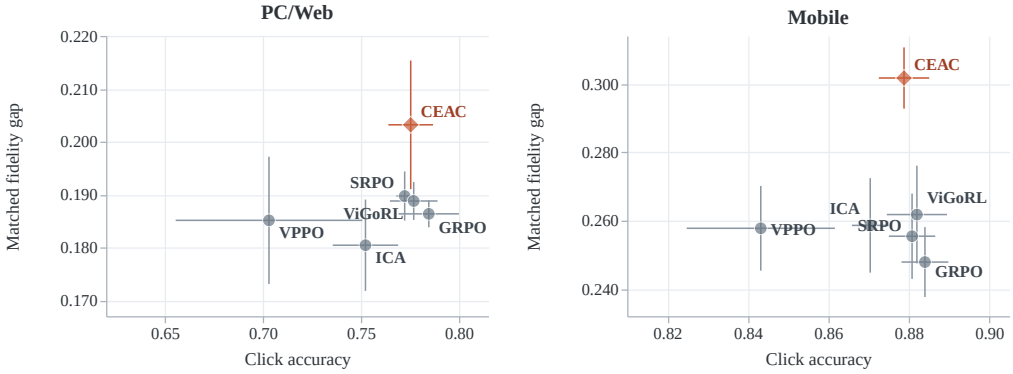


Figure 3: Accuracy and target specificity. Points are means over five seeds; bars are one sample SD. CEAC has the largest matched gap on both splits, most clearly on mobile, with no accuracy collapse.

Table 1 reports task success and target specificity separately. CEAC reaches 0.775 ± 0.011 accuracy on PC/Web and 0.879 ± 0.006 on mobile. It is more accurate than VPPO on both splits (PC/Web difference $+0.0723$, 95% CI $[+0.0348, +0.1191]$; mobile $+0.0356$, $[+0.0200, +0.0547]$; both $p_{\text{Holm}} < 0.0001$), and no other accuracy difference is significant. CEAC therefore keeps accuracy comparable to the other objectives but does not raise it over every comparator.

Figure 3 plots task accuracy against target specificity, because a larger gap would mean little if accuracy collapsed. CEAC has the largest observed gap on both splits: 0.203 on PC/Web, compared

with 0.180 to 0.190 for the other objectives, and 0.302 on mobile, versus 0.248 to 0.262. Its accuracy stays within the range of the others.

6.2 Reliability across seeds and data sources

We first check on PC/Web that CEAC's lead in mean gap holds in paired comparisons and does not hide opposite results on desktop PC and web examples.

On PC/Web, CEAC's paired gap differences remain significant after Holm correction against ICA ($+0.0228$, $p_{\mathrm{Holm}} = 0.0030$), ViGoRL ($+0.0144$, $p_{\mathrm{Holm}} = 0.0444$), and GRPO ($+0.0168$, $p_{\mathrm{Holm}} = 0.0160$), but not against SRPO or VPPO.

Table 2 splits the PC/Web result into its two sources, desktop PC and web.

CEAC has the largest gap on both desktop PC (0.232) and web (0.176). The accuracy ranking differs between the two sources, and CEAC is not the most accurate method on either. We therefore base statistical tests on the pooled PC/Web split and use the per-source rows only to show where the gap gain appears.

Table 2: PC/Web results by source, with 63 desktop PC and 67 web examples per seed. Values are mean±sample SD. The per-source rows show where the pooled result comes from; statistical tests use the pooled PC/Web split.

Objective	PC acc.	PC gap	Web acc.	Web gap
CEAC	0.749±0.024	**0.232±0.018**	0.800±0.008	**0.176±0.007**
ICA	0.727±0.017	0.206±0.015	0.776±0.018	0.156±0.007
SRPO	0.743±0.007	0.214±0.007	0.800±0.008	0.167±0.009
VPPO	0.686±0.045	0.201±0.011	0.719±0.065	0.170±0.018
ViGoRL	0.746±0.019	0.214±0.011	0.806±0.011	0.165±0.008
GRPO	0.752±0.024	0.210±0.009	**0.815±0.008**	0.164±0.005

Paired mobile gains. Table 3 tests CEAC's mobile gap advantage with comparisons paired by seed. CEAC increases the mean matched gap by 15.3% to 21.7% over each comparator. All five paired differences remain significant after Holm correction, and CEAC's gap is higher in every one of the five seeds.

The intervals and tests are computed on absolute gap differences. The percentages are relative gains in the gap, not accuracy changes in percentage points.

Table 3: Paired mobile gains in matched gap. Relative gain is $(\mathrm{FG}_{\mathrm{CEAC}} - \mathrm{FG}_{\mathrm{comp}})/\mathrm{FG}_{\mathrm{comp}}$; intervals are for the absolute gap difference. Holm correction covers five comparisons. Pos. counts the seeds in which CEAC's gap is higher.

Comparator	Rel. gain	95% CI (abs.)	p_{Holm}	Pos.
ICA	+16.7%	[0.0289, 0.0557]	< 0.0001	5/5
SRPO	+18.1%	[0.0302, 0.0626]	< 0.0001	5/5
VPPO	+17.1%	[0.0306, 0.0578]	< 0.0001	5/5
ViGoRL	+15.3%	[0.0214, 0.0573]	< 0.0001	5/5
GRPO	+21.7%	[0.0422, 0.0652]	< 0.0001	5/5

6.3 Target response and agreement between probes

A larger gap can come from a larger target response or from a smaller control response. Table 4 separates the two. From PC/Web to mobile, CEAC's target response rises by 0.0967, while its control response changes by only -0.0018. Almost all of the $+0.0985$ gap increase therefore comes from the target response.

Table 4: CEAC's target and control responses, averaged over mask and blur probes. The increase on mobile comes from the target response; the control response barely changes.

Split	Target	Control	Matched gap
PC/Web	0.2169±0.0111	0.0137±0.0021	0.2032±0.0122
Mobile	0.3136±0.0100	0.0119±0.0053	**0.3017±0.0089**
Change	+0.0967	−0.0018	+0.0985

Mask and blur, measured separately, give nearly the same mobile gaps (0.3033 and 0.3001). Together with Figure 3, the decomposition shows that CEAC's larger mobile gap does not come from a collapse in accuracy or from a general sensitivity to any edit.

The same decomposition explains CEAC's advantage over the other objectives (Table S6). Its mean mobile target response is 0.314, above the comparator range $[0.257, 0.279]$, whereas its control response of 0.012 lies within their $[0.009, 0.021]$ range. The advantage therefore comes from a

larger response when the referred element is edited. It does not come from an unusually small control response, which a policy that barely moves under any edit would also show.

Agreement across mask and blur. CEAC is trained with masks, so evaluating only with masks could favour it. Table 5 therefore also evaluates with blur. CEAC's mobile gaps are 0.3033 under mask and 0.3001 under blur, versus at most 0.2628 for any comparator under either probe, so CEAC leads by at least 0.0373. CEAC also has the highest gap on PC/Web under both probes. For every objective, the mask and blur gaps differ by at most 0.0072 on PC/Web and 0.0044 on mobile.

Table 5: CEAC has the largest gap under both mask and blur on both domains. Mean gaps differ between probes by at most 0.0072 (PC/Web) and 0.0044 (mobile).

	PC/Web gap		Mobile gap	
Objective	**Mask**	**Blur**	**Mask**	**Blur**
CEAC	**0.2068**	**0.1996**	**0.3033**	**0.3001**
ICA	0.1801	0.1808	0.2584	0.2587
SRPO	0.1885	0.1909	0.2576	0.2532
VPPO	0.1857	0.1845	0.2566	0.2588
ViGoRL	0.1896	0.1880	0.2607	0.2628
GRPO	0.1851	0.1876	0.2487	0.2470

Blur changes the region's appearance without leaving the blank patch that the training mask leaves. The agreement between probes therefore argues against an advantage that exists only under mask evaluation. Both edits are artificial and use the same regions, however, so the agreement says nothing about whether the policy uses the element's meaning.

6.4 Effect of the credit, design choices and cost

Table 6 asks whether evidence credit changes which sampled actions are reinforced or only nudges their advantages. Rank change is the share of action pairs in a group whose order differs from GRPO's, and sign change is the share of actions whose advantage has the opposite sign to GRPO's.

Table 6: Credit changes and training cost for the same 30 confirmation runs. Rank and sign changes are measured against strict GRPO within each sampled group, so GRPO scores zero. CEAC's clip fraction, the share of advantages at the clipping limit, is $2.94 \pm 0.14\%$ (all rows: from 0.65% to 3.21%). VPPO's approximate KL signals unstable training and is not a quality metric.

Objective	Time (h)	Approx. KL	Rank Δ (%)	Sign Δ (%)
CEAC	11.11±1.18	0.099±0.012	**86.36±0.63**	77.22±1.16
ICA	16.01±1.12	0.516±0.151	69.38±0.46	79.91±0.26
SRPO	7.78±0.55	0.025±0.010	0.60±0.09	0.88±0.10
VPPO	18.23±0.88	1.24e3 ± 1.76e3	70.79±1.08	80.05±1.77
ViGoRL	13.46±1.65	0.040±0.008	9.49±0.47	15.08±0.86
GRPO	8.33±0.23	0.024±0.007	0.00	0.00

CEAC changes the order of 86.4% of sampled action pairs and the sign of 77.2% of advantages, and only 2.94% of combined advantages reach the clipping limit. The evidence term therefore changes which actions are reinforced, and it is neither negligible nor mostly removed by clipping. ICA and VPPO also make large changes but do not match CEAC's combination of accuracy and gap on mobile, so the size of the change alone does not explain the result.

For each sampled action, CEAC runs three extra forward passes without gradients, with the target or one of the controls masked. These passes add training compute, but at inference the model runs once on the unedited screenshot; mask and blur passes serve only to measure the gap. Sharing computation across these passes, or computing the contrast less often, could reduce this cost.

Development choices. Table 7 answers two questions with development runs on the validation split, separate from the confirmation runs. The first is which edit should provide the training credit: mask, blur, or both (*dual credit*). The second is whether CEAC's larger gap over GRPO persists after three epochs, and at what cost in accuracy and KL.

Mask credit reaches 0.784 validation accuracy, compared with 0.768 for blur and 0.776 for dual credit, and the three give similar matched gaps. These results support training with masks and keeping blur as an independent evaluation probe. After three epochs, CEAC's gap exceeds GRPO's by 0.0280, with 0.0052 lower validation accuracy and a higher approximate KL.

Table 7: Choice of training edit (operator: mask, blur, or both as dual credit) and a three-epoch comparison between CEAC and GRPO on the development split (three seeds per cell). Bold marks the highest accuracy and gap in each block and the lower KL in the three-epoch comparison.

Comparison	Setting	Val. acc.	Gap	Approx. KL
Operator	Mask	$\mathbf{0.784 \pm 0.005}$	0.205 ± 0.011	0.254 ± 0.155
	Blur	0.768 ± 0.012	0.205 ± 0.011	0.184 ± 0.022
	Dual	0.776 ± 0.009	$\mathbf{0.208 \pm 0.004}$	0.162 ± 0.015
	GRPO	0.773 ± 0.000	0.195 ± 0.007	0.063 ± 0.043
3 epochs	CEAC	0.802 ± 0.012	$\mathbf{0.210 \pm 0.004}$	0.121 ± 0.017
	GRPO	$\mathbf{0.807 \pm 0.012}$	0.182 ± 0.008	$\mathbf{0.045 \pm 0.014}$

An appendix sweep shows that accuracy, gap, and approximate KL do not all improve as the number of controls or the evidence weight grows. All confirmation runs use two controls and $\lambda = 0.25$.

7 Discussion and Limitations

Interpreting the gap. Two successful clicks can rely on different image regions, and a large response to edits can reflect fragility rather than useful grounding. Reporting accuracy and matched gap together, as in Figure 3, separates task success from target specificity. Splitting the gap into target and control responses then separates response to the target from general sensitivity to edits. When comparing models, we suggest checking task performance first and then comparing target and control responses under the same probes. Two policies with similar click accuracy can still differ in how specifically their clicks depend on the referred element. CEAC uses this difference to assign training credit; the deployed policy still predicts from the original screenshot alone.

Scope of the evidence. The paired gains, the target–control decomposition, the probe agreement, and the analysis of credit changes support a claim about behaviour. From PC/Web training to mobile testing, matched evidence credit increases the target-specific response with no statistically significant accuracy loss against any comparator. The link between token support in training and click movement in evaluation is observed empirically rather than derived. The results say nothing about semantic understanding, and a larger gap, KL, or policy change is not always beneficial.

Limitations. The controls match the target's area and aspect ratio but not its texture, visual salience, or meaning. Mask and blur are artificial edits, and four fixed controls cannot cover every irrelevant location. An edit may move a correct click away, and a click may rightly stay in place when the same information also appears elsewhere on the screen. Natural edits, closer appearance matching, and separate responses for correct and incorrect clicks would sharpen the measure.

Five seeds say little about rare seed-level outcomes, and the evidence is weaker on PC/Web, where two gap differences are not significant after correction. Non-significant accuracy differences do not show that two methods are equally accurate. CEAC needs three extra forward passes per sampled action, and we have not tested it beyond one 3B backbone, one training source, one mobile split, and single-step clicks. Matched controls can extend to other actions with evidence in a known screen region, but longer GUI tasks also need counterfactuals over time; this study covers only the spatial part.

8 Conclusion

CEAC assigns credit in GUI grounding by comparing how much an action depends on the target with how much it depends on matched regions outside it. Across 30 confirmation runs, its PC/Web gap is significantly larger than that of three of five comparators after correction, and its mobile gap is 15.3% to 21.7% larger than that of all five, at comparable accuracy. The gain comes from the target response and persists under blur, which is not used in training. CEAC thus makes target specificity part of the training signal without changing the shared policy update. We recommend reporting the

matched gap together with accuracy, because a reward can decide whether an action succeeds and still say nothing about the visual evidence behind it.

AI USE STATEMENT

AI tools assisted with drafting and polishing the manuscript and with retrieving and discovering related literature. The authors reviewed and verified all resulting text and references and take responsibility for the final manuscript.

ETHICS STATEMENT

This work uses existing GUI grounding data and does not involve new human-subject data collection. Raw screenshots are not redistributed in the anonymous artifact. The matched-gap analysis is intended to diagnose visual evidence use; it should not be interpreted as establishing semantic understanding or as a guarantee of safe deployment.

REPRODUCIBILITY STATEMENT

Section 5 specifies the model, data splits, optimisation settings, seeds, controls, probes, and statistical procedure. The appendix provides the complete seed-level results, objective definitions, paired analyses, integrity checks, and table reproduction steps. The anonymous supplementary artifact contains the manifests, result summaries, checksums, and aggregation code needed to reproduce all reported tables and tests without model weights or restricted raw screenshots.

REFERENCES

Julius Adebayo, Justin Gilmer, Michael Muelly, Ian Goodfellow, Moritz Hardt, and Been Kim. Sanity checks for saliency maps. In *Advances in Neural Information Processing Systems*, volume 31, 2018.

Hao Bai, Yifei Zhou, Mert Cemri, Jiayi Pan, Alane Suhr, Sergey Levine, and Aviral Kumar. DigiRL: Training in-the-wild device-control agents with autonomous reinforcement learning. In *Advances in Neural Information Processing Systems*, volume 37, 2024.

Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo 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, Jiabo Ye, Xi Zhang, Tianbao Xie, Zesen Cheng, Hang Zhang, Zhibo Yang, Haiyang Xu, and Junyang Lin. Qwen2.5-VL technical report. *arXiv preprint arXiv:2502.13923*, 2025.

Kanzhi Cheng, Qiushi Sun, Yougang Chu, Fangzhi Xu, Yantao Li, Jianbing Zhang, and Zhiyong Wu. SeeClick: Harnessing GUI grounding for advanced visual GUI agents. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, pp. 9313–9332, 2024.

Xiang Deng, Yu Gu, Boyuan Zheng, Shijie Chen, Samuel Stevens, Boshi Wang, Huan Sun, and Yu Su. Mind2Web: Towards a generalist agent for the web. In *Advances in Neural Information Processing Systems*, volume 36, 2023.

Bradley Efron and Robert J. Tibshirani. *An Introduction to the Bootstrap*. Chapman & Hall, New York, 1993.

Senyu Fei, Siyin Wang, Li Ji, Ao Li, Shiduo Zhang, Liming Liu, Jinlong Hou, Jingjing Gong, Xianzhong Zhao, and Xipeng Qiu. SRPO: Self-referential policy optimization for vision-language-action models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 6718–6728, 2026.

Jakob N. Foerster, Gregory Farquhar, Triantafyllos Afouras, Nantas Nardelli, and Shimon Whiteson. Counterfactual multi-agent policy gradients. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, 2018.

Ruth C. Fong and Andrea Vedaldi. Interpretable explanations of black boxes by meaningful perturbation. In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 3429–3437, 2017.

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 *International Conference on Learning Representations*, 2025.

Yash Goyal, Ziyan Wu, Jan Ernst, Dhruv Batra, Devi Parikh, and Stefan Lee. Counterfactual visual explanations. In *Proceedings of the International Conference on Machine Learning*, pp. 2376–2384, 2019.

Daya Guo, Dejian Yang, Haowei Zhang, et al. DeepSeek-R1 incentivizes reasoning in LLMs through reinforcement learning. *Nature*, 645:633–638, 2025.

Sture Holm. A simple sequentially rejective multiple test procedure. *Scandinavian Journal of Statistics*, 6(2):65–70, 1979.

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, 2024.

Sara Hooker, Dumitru Erhan, Pieter-Jan Kindermans, and Been Kim. A benchmark for interpretability methods in deep neural networks. In *Advances in Neural Information Processing Systems*, volume 32, 2019.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2022.

Siyuan Huang, Xiaoye Qu, Yafu Li, Yun Luo, Zefeng He, Daizong Liu, and Yu Cheng. Spotlight on token perception for multimodal reinforcement learning. In *International Conference on Learning Representations*, 2026.

Alon Jacovi and Yoav Goldberg. Towards faithfully interpretable NLP systems: How should we define and evaluate faithfulness? In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, pp. 4198–4205, 2020.

Hongxin Li, Jingfan Chen, Jingran Su, Yuntao Chen, Qing Li, and Zhaoxiang Zhang. AutoGUI: Scaling GUI grounding with automatic functionality annotations from LLMs. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, pp. 10323–10358, 2025a.

Kaixin Li, Ziyang Meng, Hongzhan Lin, Ziyang Luo, Yuchen Tian, Jing Ma, Zhiyong Huang, and Tat-Seng Chua. ScreenSpot-Pro: GUI grounding for professional high-resolution computer use. *arXiv preprint arXiv:2504.07981*, 2025b.

Kevin Qinghong Lin, Linjie Li, Difei Gao, Zhengyuan Yang, Shiwei Wu, Zechen Bai, 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, 2025.

Yadong Lu, Jianwei Yang, Yelong Shen, and Ahmed H. Awadallah. OmniParser for pure vision based GUI agent. *arXiv preprint arXiv:2408.00203*, 2024.

Vitali Petsiuk, Abir Das, and Kate Saenko. RISE: Randomized input sampling for explanation of black-box models. In *British Machine Vision Conference*, 2018.

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.

Gabriel Sarch, Snigdha Saha, Naitik Khandelwal, Ayush Jain, Michael J. Tarr, Aviral Kumar, and Katerina Fragkiadaki. Grounded reinforcement learning for visual reasoning. In *Advances in Neural Information Processing Systems*, volume 38, pp. 150977–151013, 2025. doi: 10.52202/085713-5049.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

Zhihong Shao, Peiyi Wang, Qihao Zhu, et al. DeepSeekMath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024.

Ronald J. Williams. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine Learning*, 8(3–4):229–256, 1992.

David H. Wolpert and Kagan Tumer. Collective intelligence, data routing and Braess' paradox. *Journal of Artificial Intelligence Research*, 16:359–387, 2002.

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 *International Conference on Learning Representations*, 2025.

Xiaobo Xia and Ruochen Luo. GUI-R1: A generalist r1-style vision-language action model for GUI agents. *arXiv preprint arXiv:2504.10458*, 2025.

Tianbao Xie, Danyang Zhang, Jixuan Chen, Xiaochuan Li, Siheng Zhao, Ruisheng Cao, et al. OS-World: Benchmarking multimodal agents for open-ended tasks in real computer environments. In *Advances in Neural Information Processing Systems*, volume 37, 2024.

Chih-Kuan Yeh, Cheng-Yu Hsieh, Arun Sai Suggala, David I. Inouye, and Pradeep K. Ravikumar. On the (in)fidelity and sensitivity of explanations. In *Advances in Neural Information Processing Systems*, volume 32, 2019.

Ke 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 *European Conference on Computer Vision*, pp. 240–255, 2024.

Xinbin Yuan, Jian Zhang, Kaixin Li, Zhuoxuan Cai, Lujian Yao, Jie Chen, Enguang Wang, Qibin Hou, Jinwei Chen, Peng-Tao Jiang, and Bo Li. SE-GUI: Enhancing visual grounding for GUI agents via self-evolutionary reinforcement learning. In *Advances in Neural Information Processing Systems*, volume 38, pp. 127658–127679, 2025. doi: 10.52202/085713-4251.

Yuqi Zhou, Sunhao Dai, Shuai Wang, Kaiwen Zhou, Qinglin Jia, and Jun Xu. GUI-G1: Understanding R1-Zero-like training for visual grounding in GUI agents. In *Advances in Neural Information Processing Systems*, volume 38, pp. 95683–95705, 2025. doi: 10.52202/085713-3201.

A Experimental Details

The appendix reports results for every seed, the target and control parts of the gap, the paired statistical tests, and integrity checks. These show whether the gap repeats across seeds, comes from the target response, holds under blur (which is not used in training), and comes with usable accuracy. We list every seed so that seed variation need not be inferred from rounded means.

Confirmation design. The 30 runs comprise six objectives and five seeds, $\{101, 211, 307, 401, 503\}$. CEAC uses masks, two matched controls, $\lambda = 0.25$, and three epochs. ICA, SRPO, VPPO, ViGoRL, and strict GRPO share the seeds, training and test records, edited regions, and evaluation edits (mask and blur); each seed forms one block of paired comparisons.

Separation from development. The control-count and weight sweep (Section C) uses separate development runs, excluded from every confirmation mean, interval, and test. Two controls and $\lambda = 0.25$ were chosen before confirmation. Accuracy, gap, and KL do not peak at the same setting, so optimising any one of them would give a different choice.

B METHOD AND EVALUATION

B.1 CEAC ADVANTAGE

For sampled action a_i, ℓ_i^t is its mean token log probability after masking the target. The corresponding values for the two training controls are ℓ_{ij}^c. CEAC computes

$$e_i = \frac{1}{2}\sum_{j=1}^{2} \ell_{ij}^c - \ell_i^t, \tag{S1}$$

$$E_i = \mathrm{Z}(e_i), \tag{S2}$$

$$U_i = \mathrm{Center}(q_i E_i), \tag{S3}$$

$$\widetilde{A}_i = \mathrm{Center}\big[\mathrm{Clip}(A_i^{\mathrm{task}} + 0.25U_i, -5, 5)\big]\,. \tag{S4}$$

All extra forward passes for the evidence term are detached, so no gradients flow through them. The task advantage is the z score of binary click reward within each group. For group values $v_1, \ldots, v_K$, let $\bar{v} = K^{-1}\sum_i v_i$ and $s_v = [K^{-1}\sum_i (v_i - \bar{v})^2]^{1/2}$. Then $\mathrm{Center}(v_i) = v_i - \bar{v}$ and $\mathrm{Z}(v_i) = (v_i - \bar{v})/s_v$; when $s_v < 10^{-8}$, Z returns zero for every group member. For a correct click, $q_i = 1$. For an incorrect parseable click at distance d_i from the target box, $q_i = \exp[-d_i/(0.15\ \mathrm{diag}(I))]$; an invalid click has $q_i = 0$. Here d_i is the Euclidean distance from the click to the nearest point of the box rather than to its centre. Both d_i and the image diagonal are measured in the coordinates of the resized image. The implementation adds task and evidence terms, clips, and centres the result again. It does not multiply the task advantage by an evidence-dependent weight.

Positive e_i means the target mask lowers the action's support more than the control masks do. Standardisation compares contrasts within the four-action group, and q_i scales credit by click quality (the quality gate). The two Recenter steps in Figure 2 subtract the group mean without rescaling: after quality weighting and after clipping. At evaluation, the matched gap is always reported alongside task success.

B.2 DETERMINISTIC MATCHED CONTROLS

Each control has exactly the target width and height and overlaps neither the target nor another selected control. Candidate boxes lie on a deterministic 5×5 grid. A 24×24 grayscale descriptor records patch mean, standard deviation, and local edge magnitude. Candidates are ranked by their appearance distance to the target plus a term that favours spread-out centres; ties are broken by grid position.

Training uses the first two eligible controls. The evaluator uses four controls and applies the same selected regions to every method. Two of the original 512 PC/Web training records and two of the original 502 mobile records (0.39% and 0.40%) had no room for four valid controls of the same size. We removed them before the confirmation runs. The resulting splits contain 510 training and 500 mobile records; the 128 validation and 130 PC/Web test records are unchanged. No examples are excluded during a run.

Matching the geometry keeps edit sizes equal, and the number of controls does not depend on the method, even near screen edges. Ranking by appearance gives harder controls than arbitrary empty regions, and deterministic tie-breaking keeps the measured quantity fixed across evaluations. The controls represent irrelevant regions rather than semantic counterparts of the target, so the gap measures location-specific response without covering every possible edit outside the target.

B.3 EVALUATION MEASURE

For every method and operator $O \in \{\mathrm{mask}, \mathrm{blur}\}$,

$$\Delta_R^O = \frac{\|\hat{p}(I) - \hat{p}(O_R(I))\|_2}{\mathrm{diag}(I)}. \tag{S5}$$

The gap for each operator (edit type) subtracts the average response to the four shared evaluation controls,

$$\mathrm{FG}^{O} = \Delta^{O}_{\star} - \frac{1}{4}\sum_{j=1}^{4} \Delta^{O}_{cj}, \qquad \mathrm{FG} = \tfrac{1}{2}(\mathrm{FG}^{\mathrm{mask}} + \mathrm{FG}^{\mathrm{blur}}). \tag{S6}$$

Mask is the only edit used in CEAC training. Mask and blur are evaluation probes applied uniformly to all six objectives.

Subtracting the control response separates the target response from general sensitivity to edits, because the controls receive equally sized edits elsewhere. Agreement between mask and blur tests whether the results depend on how the edit looks. CEAC never trains with blur, so this check uses an edit that the objective never optimised.

Credit configurations. Table S1 states the credit signal attached to each objective. The optimizer, model, data order, and evaluation procedure are shared; objectives differ only in the information they use to distinguish actions within a training group. CEAC is the only row whose additional credit explicitly contrasts the referred target with geometrically matched non-target regions.

Table S1: Credit definitions used in the 30-run comparison. All methods use the same clipped PPO update and KL penalty to the base policy.

Objective	Credit assigned to each action	Weight
CEAC	Task advantage plus matched-mask credit U_i (standardise, quality-gate, then centre)	0.25
ICA	Click reward plus the positive standardised drop in log probability under a global central mask	1.0
SRPO	Click reward plus credit for a valid coordinate and format	0.3
VPPO	Click reward plus the positive standardised drop in log probability under global blur	0.5
ViGoRL	Click reward plus a success bonus after cropping and predicting again; crop fraction 0.3	0.5
GRPO	Binary click reward standardised within the group	None

Accuracy and gap need not rank methods in the same order. Credit for a valid coordinate or format can make actions valid more often without identifying their visual support, whereas credit based on image edits can change sensitivity to the image without changing the binary reward. A larger gap is therefore meaningful only at comparable accuracy.

Common optimization and data scale. Table S2 lists the 30-run settings. Replication comes from five independent training seeds, not repeated checkpoint evaluation. PC/Web is the interface family used for training; mobile changes layout density, aspect ratio, iconography, and interaction conventions while retaining click grounding.

Validation is used only for development. Every confirmation result uses the stated three-epoch checkpoint, without per-method seed or test-result selection, so no selection rule can account for differences between objectives.

C Development Sensitivity

The development sweep asks whether more controls or a larger evidence term uniformly improves CEAC. It is separate from the five-seed confirmation study. Every setting in the sweep uses seeds $\{11, 23, 37\}$, mask credit, and two epochs. The control-count sweep fixes $\lambda = 0.5$, and the weight sweep fixes the number of controls at two, so the two-control row and the $\lambda = 0.5$ row report the same configuration. The confirmation runs instead use two controls, $\lambda = 0.25$, three epochs, and the five seeds in Table S2.

The three metrics do not move together. Increasing the number of controls from zero to four raises the mean gap only from 0.204 to 0.208, while validation accuracy falls from 0.789 to 0.771. The two-control condition lies between these endpoints on both measures. Across $\lambda \in$

Table S2: Common training and evaluation settings. The validation split is not used to select confirmation seeds, checkpoints, or reported test results.

Setting	Value
Backbone	Qwen2.5-VL-3B-Instruct
Adapter	LoRA rank 16, alpha 32, dropout 0.05
Training examples	510 PC/Web
Confirmation examples	130 PC/Web + 500 mobile
Training seeds	101, 211, 307, 401, 503
Epochs / group size	3 / 4
Learning rate	10^{-5}
PPO passes / clip	2 / 0.2
Reference KL coefficient	0.04
CEAC training operator / controls	mask / 2
CEAC weight / advantage clip	0.25 / 5
Evaluation operators / controls	mask and blur / 4

Table S3: Effect of control count and evidence weight on the development split. All settings use mask credit, two epochs, and seeds 11, 23, and 37. The Controls rows fix $\lambda = 0.5$; the Weight rows use two controls. Accuracy, gap, and approximate KL do not improve together as the number of controls or the weight increases. The five-seed confirmation runs use two controls and $\lambda = 0.25$, chosen in advance.

Axis	Setting	Val. acc.	Gap	Approx. KL
Controls	0	0.789±0.000	0.204±0.009	0.205±0.035
	2	0.784±0.005	0.205±0.011	0.254±0.155
	4	0.771±0.023	0.208±0.007	0.185±0.020
Weight λ	0.25	0.784±0.005	0.208±0.004	0.097±0.016
	0.50	0.784±0.005	0.205±0.011	0.254±0.155
	0.75	0.792±0.012	0.204±0.013	0.303±0.027
	1.00	0.768±0.023	0.209±0.011	0.412±0.065

$\{0.25, 0.50, 0.75, 1.00\}$, the mean gap stays between 0.204 and 0.209, whereas approximate KL rises from 0.097 to 0.412 and the best accuracy occurs at $\lambda = 0.75$.

Increasing either factor does not simply help. More controls change the set of regions that the target is compared with, and a larger λ gives the evidence term more weight relative to task credit. Since the gap varies little while KL varies a lot, heavier weighting mainly moves the policy away from its reference. Confirmation keeps the two controls and $\lambda = 0.25$ chosen beforehand instead of picking the best setting after seeing the sweep.

With one backbone and three seeds per setting, these one-factor changes cannot show that CEAC is insensitive in general. They do show that the confirmation setting sits in a stable part of the observed range, and the final claim does not rely on a setting chosen after the fact.

D Results for All Five Seeds

An aggregate can hide two problems. A single favourable seed may drive the mean, or a cross-domain gain may appear only after source and target interfaces are pooled. Tables S4 and S5 therefore list all 60 method, seed, and domain summaries behind the main-paper means. Every PC/Web row uses 130 unique evaluation records and every mobile row uses 500.

Source-interface variation. On PC/Web, CEAC's matched gap ranges from 0.1939 to 0.2242 and remains positive under both operators for every seed. In matched-seed comparisons it exceeds ICA, ViGoRL, and GRPO in all five seeds, and exceeds SRPO and VPPO in four of five. The single-seed reversals against SRPO and VPPO are small, but they are enough to make the paired tests, after correction for multiple comparisons, not significant for those two comparators, as quantified later in Table S9.

Table S4: Complete PC/Web confirmation results. Every row contains the same 130 evaluation records; matched gap averages the mask and blur gaps.

Method	Seed	Accuracy	Mask gap	Blur gap	Matched gap
CEAC	101	0.7615	0.1923	0.2009	0.1966
CEAC	211	0.7923	0.2005	0.1965	0.1985
CEAC	307	0.7769	0.2103	0.1951	0.2027
CEAC	401	0.7692	0.2338	0.2145	0.2242
CEAC	503	0.7769	0.1970	0.1909	0.1939
ICA	101	0.7615	0.1933	0.1890	0.1912
ICA	211	0.7538	0.1761	0.1806	0.1783
ICA	307	0.7231	0.1738	0.1737	0.1737
ICA	401	0.7615	0.1910	0.1842	0.1876
ICA	503	0.7615	0.1663	0.1764	0.1714
SRPO	101	0.7769	0.1847	0.1837	0.1842
SRPO	211	0.7692	0.1918	0.1925	0.1921
SRPO	307	0.7692	0.1839	0.1873	0.1856
SRPO	401	0.7769	0.1940	0.1892	0.1916
SRPO	503	0.7692	0.1884	0.2019	0.1951
VPPO	101	0.7000	0.1722	0.1791	0.1757
VPPO	211	0.7615	0.2088	0.1961	0.2025
VPPO	307	0.6308	0.1712	0.1743	0.1728
VPPO	401	0.7000	0.1960	0.1859	0.1909
VPPO	503	0.7231	0.1805	0.1872	0.1838
ViGoRL	101	0.7769	0.1953	0.1775	0.1864
ViGoRL	211	0.7923	0.1807	0.1882	0.1845
ViGoRL	307	0.7692	0.1933	0.1905	0.1919
ViGoRL	401	0.7846	0.1867	0.1899	0.1883
ViGoRL	503	0.7615	0.1920	0.1938	0.1929
GRPO	101	0.7692	0.1812	0.1829	0.1821
GRPO	211	0.7692	0.1837	0.1902	0.1869
GRPO	307	0.8000	0.1869	0.1909	0.1889
GRPO	401	0.7846	0.1885	0.1853	0.1869
GRPO	503	0.8000	0.1850	0.1890	0.1870

Accuracy shows a different pattern. CEAC is more accurate than VPPO in every seed, but its comparison with SRPO, ViGoRL, and GRPO changes sign across seeds. We therefore describe the PC/Web result as a larger matched gap at comparable accuracy, without claiming higher accuracy throughout.

Cross-domain reading. CEAC's mean matched gap rises from approximately 0.203 on PC/Web to 0.302 on mobile. The corresponding increase is not confined to one seed: the lowest mobile CEAC gap remains well above the highest PC/Web CEAC gap. The same checkpoints are evaluated on both domains, so the comparison reflects a change in response when the interface changes; no mobile training is involved.

Interface-shift variation. On mobile, CEAC's matched gap ranges from 0.2934 to 0.3156. It exceeds every comparator in every one of the 25 same-seed comparisons. Mask and blur both place CEAC first in every same-seed comparison, even though only mask appears in CEAC training. This agreement across seeds and probes makes the mobile effect more reliable than a pooled mean alone would suggest.

Mobile accuracy remains close to ICA, SRPO, ViGoRL, and GRPO and is higher than VPPO for all five seeds. Some comparator seeds are slightly more accurate than CEAC, so we make no claim of higher click accuracy on mobile. The full matrix does show a repeatable increase in target-specific response, and no accuracy collapse that could trivially inflate the gap.

Table S5: Complete mobile confirmation results. Every row contains the same 500 evaluation records; matched gap averages the mask and blur gaps.

Method	Seed	Accuracy	Mask gap	Blur gap	Matched gap
CEAC	101	0.8840	0.2942	0.2943	0.2942
CEAC	211	0.8800	0.3181	0.3132	0.3156
CEAC	307	0.8800	0.2978	0.2890	0.2934
CEAC	401	0.8680	0.3035	0.3006	0.3020
CEAC	503	0.8820	0.3028	0.3035	0.3031
ICA	101	0.8720	0.2745	0.2742	0.2744
ICA	211	0.8720	0.2654	0.2742	0.2698
ICA	307	0.8640	0.2411	0.2383	0.2397
ICA	401	0.8680	0.2543	0.2556	0.2549
ICA	503	0.8760	0.2568	0.2513	0.2541
SRPO	101	0.8860	0.2698	0.2704	0.2701
SRPO	211	0.8800	0.2477	0.2478	0.2478
SRPO	307	0.8720	0.2555	0.2370	0.2463
SRPO	401	0.8800	0.2668	0.2691	0.2680
SRPO	503	0.8860	0.2483	0.2417	0.2450
VPPO	101	0.8180	0.2474	0.2342	0.2408
VPPO	211	0.8580	0.2542	0.2599	0.2570
VPPO	307	0.8500	0.2520	0.2603	0.2562
VPPO	401	0.8300	0.2539	0.2639	0.2589
VPPO	503	0.8600	0.2755	0.2759	0.2757
ViGoRL	101	0.8780	0.2873	0.2867	0.2870
ViGoRL	211	0.8840	0.2494	0.2545	0.2519
ViGoRL	307	0.8840	0.2551	0.2592	0.2571
ViGoRL	401	0.8920	0.2570	0.2580	0.2575
ViGoRL	503	0.8720	0.2547	0.2555	0.2551
GRPO	101	0.8800	0.2319	0.2342	0.2330
GRPO	211	0.8840	0.2605	0.2578	0.2592
GRPO	307	0.8820	0.2558	0.2551	0.2555
GRPO	401	0.8800	0.2486	0.2457	0.2472
GRPO	503	0.8940	0.2467	0.2420	0.2444

E TARGET AND CONTROL DECOMPOSITION

The matched gap can increase either because target edits move the prediction more or because control edits move it less. Only the first explanation is consistent with the intended mechanism. Table S6 therefore decomposes each five-seed mean before subtraction.

Table S6: Target and control decomposition across five seeds for every objective. Target and control average the mask and blur components before seed aggregation; operator-specific gaps appear in Tables S4 and S5.

Split	Method	Target	Control	Matched gap
PC/Web	CEAC	0.217±0.011	0.014±0.002	0.203±0.012
	ICA	0.202±0.008	0.022±0.004	0.180±0.009
	SRPO	0.204±0.005	0.014±0.002	0.190±0.005
	VPPO	0.211±0.017	0.026±0.012	0.185±0.012
	ViGoRL	0.200±0.004	0.011±0.002	0.189±0.004
	GRPO	0.199±0.003	0.013±0.001	0.186±0.003
Mobile	CEAC	0.314±0.010	0.012±0.005	0.302±0.009
	ICA	0.270±0.015	0.011±0.002	0.259±0.014
	SRPO	0.265±0.013	0.009±0.001	0.255±0.012
	VPPO	0.279±0.011	0.021±0.015	0.258±0.012
	ViGoRL	0.271±0.015	0.010±0.002	0.262±0.014
	GRPO	0.257±0.011	0.009±0.002	0.248±0.010

On PC/Web, CEAC has target response 0.217, compared with 0.211 for the next-largest value, while its control response 0.014 lies inside the comparison range $[0.011, 0.026]$. On mobile, the difference is clearer: CEAC's target response is 0.314, the next-largest value is 0.279, and its control response 0.012 again lies inside the comparison range $[0.009, 0.021]$. An unusually insensitive control term therefore cannot explain the larger matched gap.

The domain shift points the same way. From PC/Web to mobile, CEAC's target response increases by about 0.097, whereas its control response changes by about -0.002. Most of the roughly 0.099 increase in matched gap is thus attributable to the target term. Without the two components, a larger gap could equally reflect a generic drop in edit sensitivity.

Operator and seed decomposition. Table S7 retains the mask and blur components for every CEAC seed. It checks whether the aggregate explanation holds for every seed and for the edit not used during training.

Table S7: CEAC decomposition for every seed and probe. Both mask and blur locate the increase from PC/Web to mobile in target response.

Split	Probe	Seed	Target	Control	Gap
PC/Web	Mask	101	0.2105	0.0182	0.1923
		211	0.2163	0.0157	0.2005
		307	0.2233	0.0131	0.2103
		401	0.2459	0.0121	0.2338
		503	0.2106	0.0136	0.1970
	Blur	101	0.2169	0.0159	0.2009
		211	0.2095	0.0130	0.1965
		307	0.2060	0.0109	0.1951
		401	0.2262	0.0117	0.2145
		503	0.2038	0.0129	0.1909
Mobile	Mask	101	0.3080	0.0138	0.2942
		211	0.3276	0.0095	0.3181
		307	0.3057	0.0079	0.2978
		401	0.3258	0.0223	0.3035
		503	0.3116	0.0088	0.3028
	Blur	101	0.3063	0.0121	0.2943
		211	0.3219	0.0087	0.3132
		307	0.2971	0.0081	0.2890
		401	0.3198	0.0192	0.3006
		503	0.3118	0.0084	0.3035

Every one of the 20 operator-seed rows has a positive gap. On PC/Web, the largest mask-versus-blur difference occurs for seed 401, but both probes still attribute most displacement to the target. On mobile, the two probe gaps are close for every seed and the control terms remain small relative to target response. Agreement under blur does not imply invariance to arbitrary image edits, but it excludes the possibility that CEAC scores well only because evaluation reuses its training mask.

These results concern location-specific dependence on evidence; they do not recover a causal model of the interface. They come from one 3B backbone, single-step click grounding, deterministic controls, and mask and blur interventions. Natural edits, learned control matching, longer trajectories, other model scales, and independently collected interface families would each need separate validation. The gap must still be read with task accuracy and the target-control decomposition.

F Complete Paired Inference

Seed tables show the direction of each difference but not the uncertainty at the screenshot level, so the paired analysis models both sources of variation. Differences are CEAC minus comparator. Each confidence interval uses 10,000 paired hierarchical resamples: sample five seed blocks with replacement, then sample screenshot groups with replacement within each selected seed. Pairing

keeps the method comparison on the same examples and perturbation geometry. Unadjusted two-sided p-values are twice the smaller of the two bootstrap tail probabilities (difference below or above zero). Holm correction is applied across the five comparators separately for each split and metric.

F.1 PC/WEB INFERENCE

Tables S8 and S9 report task success and matched gap separately on PC/Web, the training interface family.

Table S8: Paired PC/Web accuracy inference. CEAC is significantly more accurate than VPPO; the other comparisons are unresolved after Holm correction.

Comparator	Δ accuracy	95% CI	p	p_{Holm}	+ seeds
ICA	$+0.0231$	$[-0.0015, +0.0505]$	0.0754	0.3016	4/5
SRPO	$+0.0031$	$[-0.0138, +0.0196]$	0.7752	1.0000	3/5
VPPO	$+0.0723$	$[+0.0348, +0.1191]$	< 0.0001	< 0.0001	5/5
ViGoRL	-0.0015	$[-0.0185, +0.0154]$	0.9284	1.0000	2/5
GRPO	-0.0092	$[-0.0294, +0.0119]$	0.4162	1.0000	1/5

Table S9: Paired PC/Web matched gap inference. CEAC exceeds ICA, ViGoRL, and GRPO after Holm correction; SRPO and VPPO remain unresolved.

Comparator	Δ matched gap	95% CI	p	p_{Holm}	+ seeds
ICA	$+0.0228$	$[+0.0089, +0.0379]$	0.0006	0.0030	5/5
SRPO	$+0.0135$	$[+0.0005, +0.0279]$	0.0416	0.0832	4/5
VPPO	$+0.0181$	$[+0.0001, +0.0361]$	0.0494	0.0832	4/5
ViGoRL	$+0.0144$	$[+0.0024, +0.0287]$	0.0148	0.0444	5/5
GRPO	$+0.0168$	$[+0.0047, +0.0313]$	0.0040	0.0160	5/5

For accuracy, only the comparison with VPPO remains significant after correction; the intervals for ICA, SRPO, ViGoRL, and GRPO include zero. For matched gap, CEAC exceeds ICA by 0.0228, ViGoRL by 0.0144, and GRPO by 0.0168 after Holm correction. The positive estimates against SRPO and VPPO do not survive correction. On the source domain, CEAC thus improves the gap over three of five comparators, and its accuracy differences from four of the five are not statistically significant.

F.2 MOBILE INFERENCE

The mobile tables repeat the same paired comparisons on mobile screens; they use the same training runs, not a different set.

Table S10: Paired mobile accuracy inference. CEAC is significantly more accurate than VPPO; the other comparisons are unresolved after Holm correction.

Comparator	Δ accuracy	95% CI	p	p_{Holm}	+ seeds
ICA	$+0.0084$	$[-0.0004, +0.0175]$	0.0682	0.2728	4/5
SRPO	-0.0020	$[-0.0099, +0.0056]$	0.6676	1.0000	1/5
VPPO	$+0.0356$	$[+0.0200, +0.0547]$	< 0.0001	< 0.0001	5/5
ViGoRL	-0.0032	$[-0.0155, +0.0079]$	0.6420	1.0000	2/5
GRPO	-0.0052	$[-0.0127, +0.0020]$	0.1842	0.5526	1/5

The accuracy result mirrors PC/Web: CEAC exceeds VPPO, while the other four comparisons are not significant after Holm correction. The matched-gap result is stronger. All five comparisons remain significant after Holm correction, and their confidence intervals exclude zero, with gains from 0.0399 over ViGoRL to 0.0538 over GRPO. All five seed-level differences are positive for every comparator. For this reason the main paper emphasises mobile. The change to mobile interfaces enlarges the gap difference, although CEAC does not gain accuracy over every method.

Table S11: Paired mobile matched gap inference. CEAC exceeds all five comparators after Holm correction.

Comparator	Δ matched gap	95% CI	p	p_{Holm}	+ seeds
ICA	+0.0431	[+0.0289, +0.0557]	< 0.0001	< 0.0001	5/5
SRPO	+0.0463	[+0.0302, +0.0626]	< 0.0001	< 0.0001	5/5
VPPO	+0.0440	[+0.0306, +0.0578]	< 0.0001	< 0.0001	5/5
ViGoRL	+0.0399	[+0.0214, +0.0573]	< 0.0001	< 0.0001	5/5
GRPO	+0.0538	[+0.0422, +0.0652]	< 0.0001	< 0.0001	5/5

Scope of the intervals. Hierarchical resampling avoids treating the 630 screenshots within a seed as 630 independent training replications. Five training seeds remain a limitation. The intervals describe uncertainty under the observed seed and screenshot hierarchy and do not imply that the ordering holds for other backbones, datasets, or longer training schedules.

Summary across splits. No CEAC accuracy decrease is significant after Holm correction against any comparator on either split, and CEAC is more accurate than VPPO on both. Non-significant intervals still allow small gains or losses and do not show equivalence. On PC/Web, three of the five gap gains are significant; on mobile, all five are, and CEAC is ahead in every same-seed comparison. We report the domains separately so that pooling does not mask the weaker source comparisons.

G Integrity Checks

Purpose. The confirmation claim combines runs launched in separate batches and evaluated on two domains. Two rows are comparable only if they share the intended model, data split, evaluation records, and perturbation geometry, which a complete JSON file alone does not guarantee. The checks below verify this before aggregation.

Run validation. A run is accepted only when its run ID, method, seed, manifest hash, configuration hash, split hash, and source hashes match; training covers 1,530 example presentations with zero errors; the two evaluation ID sets contain exactly 130 and 500 unique records; all JSON files parse; and the terminal marker records hashes for every declared result file.

Table S12: Validation summary for the confirmation study.

Audit item	Result
Included units	30
Complete and valid units	30
Methods / seeds per method	6 / 5
CEAC training controls / weight	2 / 0.25
Common evaluation controls	4
Training errors	0
PC/Web coverage per unit	130/130
Mobile coverage per unit	500/500
JSON parse failures	0
Identity/hash conflicts	0

Cross-matrix identity checks. The CEAC runs and baseline runs were completed under separate manifests, so each run is validated against the manifest that launched it. The aggregator then verifies a common model revision, dataset revision, split file hash, evaluation record IDs, and evaluation control geometry across all 30 runs. The primary aggregate contains only the six objectives in the confirmation matrix and excludes all development runs.

H Artifact Contents and Reproduction Path

Package contents. The anonymous artifact contains code, lists of record IDs defining each split, manifests, JSON result summaries, checksums, and aggregation scripts. It excludes model weights, LoRA adapters, raw screenshots, private execution logs, signed URLs, tokens, cookies, and credentials.

Reconstructing the reported tables. First, validate each run's manifest and exact PC/Web and mobile record-ID sets. Second, recompute target and control displacement under mask and blur, averaging four controls and two operator gaps. Third, aggregate by method and seed and apply the paired hierarchical resampling in Section F. All seed, component, and paired tables are computed from the same validated per-record summaries.

Anonymous reproduction. The package supports verification of every number in Tables S4–S11: operator components, seed means, confidence intervals, and Holm correction. Re-running training also requires the stated base-model revision and underlying screenshots, neither redistributed. Reviewers can thus check the reported inference without access to restricted data or model artifacts.