CA-OPD: CONFIDENCE-AWARE ON-POLICY DISTILLATION FOR STRUCTURED VISUAL PREDICTION

Anonymous authors
Paper under double-blind review

ABSTRACT

Autoregressive vision language models unify heterogeneous perception tasks but are highly susceptible to compounding errors. On-policy distillation (OPD) bridges the training-inference mismatch by training students on their own rollouts. However, unreliable student predictions, especially early in training, can derail the trajectory and degrade the quality of teacher supervision. While recent interleaved distillation methods allow the teacher to verify and replace student tokens, they primarily rely on rigid ranking metrics rather than exact teacher confidence, and they overlook how intervention decisions can inform token-level supervision. To address this, we introduce Confidence-Aware On-Policy Distillation (CA-OPD), a framework that couples reliable rollout construction with adaptive supervision. CA-OPD utilizes teacher confidence to selectively correct unreliable student transitions, gradually transferring rollout control to the student via a strict-to-relaxed schedule. Crucially, CA-OPD aligns knowledge transfer with these intervention decisions: corrected positions receive direct cross-entropy supervision from the teacher's prediction, while retained positions benefit from the teacher's full predictive distribution. Evaluated in a multi-teacher setting for GUI grounding and optical character recognition, CA-OPD substantially improves the Qwen3.5-0.8B baseline across all six target benchmarks, including gains of 9.50 points on ScreenSpot-Pro and 6.72 points on OCRBench-v2 English. Controlled studies further show that the gains depend on intervention placement, progressive rollout control, and intervention-aligned supervision, rather than intervention frequency alone.

1 INTRODUCTION

Vision language models (VLMs) increasingly solve heterogeneous perception tasks through a unified autoregressive interface (Peng et al., 2023; Wang et al., 2026). For instance, optical character recognition (OCR) outputs are generated as text, while visual grounding and graphical user interface (GUI) interaction serialize locations or actions as token sequences (Kim et al., 2021; Chen et al., 2021; You et al., 2024). While this formulation provides a flexible, task-agnostic interface, it makes structured prediction inherently sequential: every generated token becomes part of the next input. Once an incorrect token is committed, subsequent predictions are conditioned on a corrupted prefix. A local error can therefore compound, degrading both the immediate prediction and the entire trajectory along which the remainder of the output is generated (Ranzato et al., 2015; Kim et al., 2022).

This autoregressive dependency poses a significant challenge for knowledge distillation (KD). Traditional offline and sequence-level distillation train the student on stable, reference- or teacher-generated trajectories (Hinton et al., 2015; Kim & Rush, 2016). However, during inference, the student is evaluated on prefixes produced by its own policy. On-policy distillation (OPD) mitigates this mismatch by allowing the student to generate the rollout, querying the teacher for supervision on the states the student actually visits (Agarwal et al., 2024; Gu et al., 2024; Yuan et al., 2026). Yet, the very property that aligns OPD with inference also introduces a critical vulnerability. When the student is unreliable, especially early in training, an incorrect prediction can enter the prefix and derail all subsequent states where teacher supervision is applied (Xu et al., 2025).

Recent interleaved distillation methods offer a natural remedy for this tension. Instead of blindly committing every student prediction to the rollout, the teacher can verify student proposals and intervene when necessary. Speculative Knowledge Distillation (SKD) (Xu et al., 2025), for example,

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.

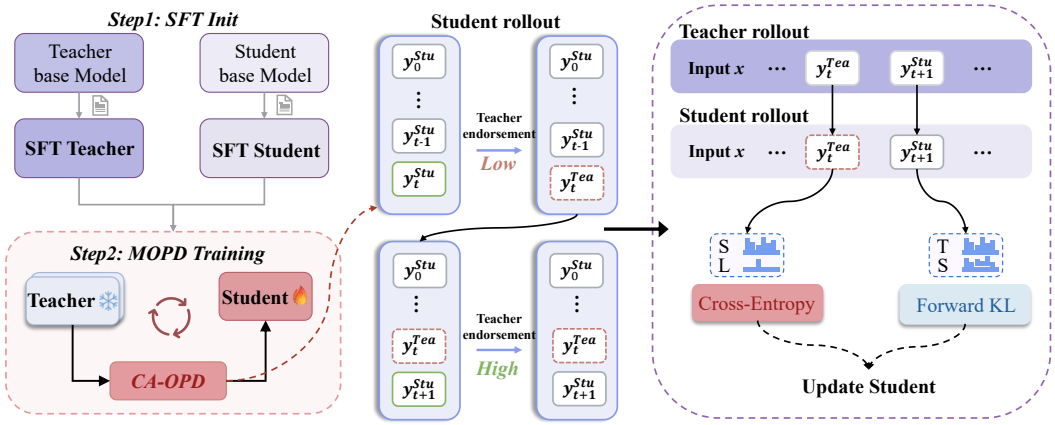

Figure 1: **Confidence-aware on-policy distillation for structured visual prediction.** CA-OPD evaluates each student proposal using teacher confidence. Low-support proposals are replaced by teacher tokens and receive direct corrective supervision, while supported proposals remain in the rollout and retain distributional supervision. The mixed prefix is used for subsequent autoregressive generation.

uses the teacher's ranking of a student proposal to decide whether the proposal should remain in the generated prefix. Such verification provides a useful middle ground between relying entirely on student generation during OPD and using trajectories dictated by the teacher. It protects the rollout from severe student errors while retaining substantial exposure to states induced by the student policy.

Nevertheless, two limitations remain. First, rank does not directly measure how strongly the teacher supports a student proposal. A token can rank highly even when the teacher distribution is uncertain, whereas proposals with the same rank can receive substantially different probability mass. The decision to trust a student transition is therefore better informed by the exact teacher confidence assigned to that proposal. Second, teacher intervention contains useful learning signals beyond simply repairing the rollout. When the teacher rejects a student proposal and commits its own prediction, it identifies a concrete local mistake. Treating this event solely as a rollout-correction mechanism leaves this corrective signal underutilized. Conversely, when the teacher already supports the student proposal, preserving the richer teacher distribution can provide informative supervision. These observations suggest that rollout control and token-level supervision should be jointly optimized.

To address this, we introduce CA-OPD, a confidence-aware on-policy distillation framework that couples reliable rollout construction with adaptive token-level supervision. CA-OPD leverages teacher confidence to assess student proposals and selectively correct unreliable transitions. Teacher intervention is stronger early in training when the student is less reliable and gradually decreases, allowing the rollout to smoothly move toward greater student control. Crucially, the outcome of this verification dictates how knowledge is transferred: *corrected positions receive direct supervision from the teacher's prediction, while retained positions continue to benefit from the teacher's full predictive distribution.* As illustrated in Figure 1, CA-OPD preserves the on-policy character of training while selectively preventing poorly supported student transitions from being committed to the prefix. This same gate determines the supervision used at each position, elegantly connecting state visitation and knowledge transfer within a single autoregressive process.

We evaluate CA-OPD in a multi-teacher setting that distills complementary GUI grounding and OCR capabilities into Qwen3.5-0.8B. Relative to the Qwen3.5-0.8B baseline, CA-OPD improves all six target benchmarks, with gains of 9.08 points on ScreenSpot-v2, 9.50 points on ScreenSpot-Pro, 6.72 points on OCRBench-v2 English, and 9.77 points on OCRBench-v2 Chinese. It also outperforms Offline KD, standard OPD, and SKD on every target benchmark, indicating that the gains cannot be attributed to distillation alone. Furthermore, controlled studies demonstrate that our gains cannot be explained by intervention frequency alone. Specifically, we show that (1) where the teacher intervenes matters, as randomly placed interventions provide minimal benefit compared to confidence-based ones at identical budgets; (2) a strict-to-relaxed schedule outperforms both fixed and reversed alternatives; and (3) aligning the supervision objective with the intervention decision materially contributes to the

overall improvement. Together, these results indicate that effective on-policy distillation depends not only on how often the teacher intervenes, but also on the strategic placement of interventions, the evolution of control over training, and the adaptive supervision of corrected versus retained positions.

Our contributions are summarized below:

- We identify a coupled challenge in on-policy distillation for structured autoregressive prediction: unreliable student transitions corrupt subsequent states conditioned for generation, and existing methods fail to utilize the teacher's intervention decision as a signal for token-level supervision.
- We introduce CA-OPD, which uses teacher confidence to selectively repair student-generated prefixes, progressively transfers rollout control to the student, and applies intervention-aligned supervision: cross-entropy at replaced positions and distributional distillation at retained positions.
- We demonstrate consistent gains across GUI grounding and OCR in a multi-teacher distillation setting while preserving held-out capabilities. Controlled studies confirm that improvements stem from strategic intervention placement, the direction of the rollout-control schedule, and adaptive supervision, rather than sheer intervention frequency.

2 Related Work

Autoregressive structured visual prediction. Representing structured visual outputs as token sequences is a prevailing paradigm for unifying heterogeneous perception tasks. For instance, Pix2Seq (Chen et al., 2021) and Donut (Kim et al., 2021) serialize bounding boxes and document texts, while Unified-IO models map diverse multimodal outputs into a shared autoregressive interface (Lu et al., 2022; 2024). Recent VLMs similarly express visual grounding and GUI interactions through language-like coordinate sequences or action tokens (Peng et al., 2023; Chen et al., 2023; You et al., 2024; Cheng et al., 2024; Lin et al., 2025). While this formulation provides a flexible, task-agnostic interface, it makes structured prediction inherently sequential. Because every committed token becomes part of the subsequent prefix, errors in OCR symbols, coordinates, or actions inevitably compound. Such error propagation is particularly consequential for structured outputs, where a small local deviation can change the semantic interpretation of the entire prediction or redirect subsequent generation toward incorrect states.

Knowledge distillation for autoregressive models. Traditional knowledge distillation (KD) transfers knowledge via teacher predictive distributions (Hinton et al., 2015), while sequence-level KD relies on complete teacher-generated trajectories (Kim & Rush, 2016). Recent advances in language and multimodal model distillation primarily broaden what knowledge is transferred or which divergence is minimized. For example, Distilling Step-by-Step leverages teacher rationales (Hsieh et al., 2023), MiniLLM optimizes reverse-KL divergence to better match generation distributions (Gu et al., 2024), and LLaVA-KD transfers complex multimodal relations (Cai et al., 2025). While these approaches enrich the available supervision, their effectiveness is bottlenecked by the autoregressive states where this supervision is applied. Because errors compound in structured prediction, static teacher-generated prefixes fail to reflect true inference trajectories. CA-OPD complements these representation-focused methods by jointly optimizing rollout construction and token-level supervision.

On-policy and interleaved distillation. The distribution shift between training and inference states is a classic challenge in sequential prediction, historically addressed by techniques like scheduled sampling (Bengio et al., 2015) and DAgger (Ross et al., 2011). In modern KD, Generalized Knowledge Distillation (GKD) addresses this by querying the teacher on student-generated rollouts, forcing the student to learn from its own inference-time errors (Agarwal et al., 2024). SKD refines this via interleaved trajectories, where the teacher replaces student proposals that rank poorly under its distribution (Xu et al., 2025). Although such interleaving improves the reliability of student-generated trajectories, a rank-based criterion only measures the relative ordering of a proposal and does not directly quantify how strongly the teacher supports it. CA-OPD refines this interleaved approach by replacing rigid ranking with the teacher's confidence to guide interventions. It dynamically relaxes this intervention threshold as the student improves, and explicitly couples rollout control with adaptive token-level supervision, thereby unifying state visitation and knowledge transfer.

3 PROBLEM FORMULATION

3.1 STRUCTURED AUTOREGRESSIVE PREDICTION

Let $x = (I, q)$ denote an input pair comprising an image I and a textual instruction q. An autoregressive model generates an output sequence $y = (y_1, \ldots, y_L)$ over a vocabulary $\mathcal{V}$ such that:

$$\pi(y \mid x) = \prod_{t=1}^{L} \pi(y_t \mid x, y_{<t}). \tag{1}$$

We define the autoregressive state at step t as $s_t = (x, y_{<t})$. Given a shared tokenizer, the student network (parameterized by θ) and the frozen teacher network define the next-token predictive distributions $\pi_S(\cdot \mid s_t; \theta)$ and $\pi_T(\cdot \mid s_t)$, respectively.

In structured visual prediction, the generated tokens may encode diverse outputs such as text, spatial coordinates, or discrete actions. Because each committed token updates the subsequent state, an incorrect prediction corrupts not only the immediate output but also the future contexts upon which downstream predictions and supervision rely.

3.2 ON-POLICY DISTILLATION AND STATE VISITATION

OPD constructs training trajectories by sampling directly from the student policy,

$$\hat{y}_t \sim \pi_S(\cdot \mid \hat{s}_t; \theta), \tag{2}$$

and aligns the student with the teacher over these actively visited states:

$$\mathcal{L}_{\mathrm{OPD}} = \frac{1}{\sum_t m_t} \sum_{t=1}^{L} m_t D_{\mathrm{KL}} \left(\pi_T(\cdot \mid \hat{s}_t) \,\|\, \pi_S(\cdot \mid \hat{s}_t; \theta) \right), \tag{3}$$

where m_t denotes the loss mask.

While this mitigates the distribution shift between training and free-running inference, unreliable student predictions, especially early in training, can severely degrade the quality of subsequent states. To address this, we explicitly decouple the token-level supervision objective from the behavior policy μ used to construct the rollout:

$$y_t \sim \mu(\cdot \mid s_t), \qquad s_{t+1} = (x, y_{\leq t}). \tag{4}$$

Standard OPD corresponds to $\mu = \pi_S$, which fully exposes the trajectory to student errors. Conversely, interleaved distillation permits teacher intervention during prefix construction. CA-OPD builds upon this generalized formulation by adaptively gating when a student proposal is retained. This dynamically balances the need for reliable, high-quality training trajectories with the necessity of gradually exposing the student to its own generated states.

4 CONFIDENCE-AWARE ON-POLICY DISTILLATION

Building on Section 3, CA-OPD constructs the rollout prefix dynamically. At each decoding step, the teacher evaluates the student's proposed token. Highly supported proposals are retained and supervised via the teacher's predictive distribution, whereas unreliable ones are replaced by the teacher's prediction and supervised via direct cross-entropy. This single intervention decision elegantly aligns rollout control with adaptive token-level supervision.

4.1 CONFIDENCE-AWARE ROLLOUT CONTROL

At state s_t, the student generates a candidate token $\hat{y}_t \sim \pi_S(\cdot \mid s_t; \theta)$. We measure the teacher's support for this exact proposal using the negative log-likelihood (NLL) defined as

$$\mathrm{NLL}_{T,t} = -\log \pi_T(\hat{y}_t \mid s_t) \tag{5}$$

A lower $\mathrm{NLL}_{T,t}$ indicates stronger teacher support. Given an NLL threshold $\tau_{\mathrm{nll},s}$ at optimization step s, we define a binary replacement indicator as

$$r_t = \mathbb{I}\left[\mathrm{NLL}_{T,t} > \tau_{\mathrm{nll},s}\right] \tag{6}$$

Here, $r_t = 1$ indicates the proposal is rejected, and $r_t = 0$ means it is retained. Equivalently, this corresponds to a teacher probability floor $\alpha_s = \exp(-\tau_{\mathrm{nll},s})$. When a proposal is rejected, the teacher provides a deterministic correction $\tilde{y}_t = \arg\max_{v \in \mathcal{V}} \pi_T(v \mid s_t)$. The final token committed to the rollout is therefore

$$y_t = \begin{cases} \tilde{y}_t, & r_t = 1, \\ \hat{y}_t, & r_t = 0. \end{cases} \tag{7}$$

Generation subsequently continues from $s_{t+1} = (x, y_{\le t})$. Because the prefix may contain both retained student tokens and teacher corrections, an intervention alters not only y_t but also the autoregressive state conditioning all future predictions. These are discrete rollout operations, and no gradients are propagated through them.

4.2 Intervention-Aligned Supervision

The replacement indicator r_t also dictates the token-level supervision applied at each position.

Supervision at replaced positions. For a replaced position ($r_t = 1$), the committed token is the teacher's prediction $y_t = \tilde{y}_t$. The student is directly supervised using standard cross-entropy as

$$\mathcal{L}_t^{\mathrm{R}} = -\log \pi_S(y_t \mid s_t; \theta) \tag{8}$$

Supervision at retained positions. For a retained position ($r_t = 0$), the student proposal remains. The student is then supervised by the teacher's predictive distribution via the forward KL divergence formulated as

$$\mathcal{L}_t^{\mathrm{K}} = D_{\mathrm{KL}}\left(\pi_T(\cdot \mid s_t) \,\|\, \pi_S(\cdot \mid s_t; \theta)\right). \tag{9}$$

To mitigate the computational cost of transferring the full vocabulary distribution, we distill only the teacher's top-k predictions $\mathcal{V}_{T,t}^{(k)} = \mathrm{TopK}_v\, \pi_T(v \mid s_t)$. The truncated forward KL term is computed as

$$\mathcal{L}_t^{\text{K-topk}} = \sum_{v \in \mathcal{V}_{T,t}^{(k)}} \pi_T(v \mid s_t)\left[\log \pi_T(v \mid s_t) - \log \pi_S(v \mid s_t; \theta)\right] \tag{10}$$

Because the retained probabilities are unnormalized, this truncated value can occasionally become negative. We therefore apply a zero lower bound where $\mathcal{L}_t^{\text{K-topk}} \leftarrow \max(\mathcal{L}_t^{\text{K-topk}}, 0)$.

4.3 CA-OPD Objective

By unifying rollout construction and supervision under the same replacement indicator, the token-level CA-OPD objective becomes

$$\mathcal{L}_t^{\text{CA-OPD}} = r_t \mathcal{L}_t^{\mathrm{R}} + (1 - r_t)\mathcal{L}_t^{\text{K-topk}} \tag{11}$$

Let $m_t \in \{0, 1\}$ denote the valid response token mask. The sequence-level objective is formulated as

$$\mathcal{L}_{\text{CA-OPD}} = \frac{1}{\sum_{t=1}^{L} m_t} \sum_{t=1}^{L} m_t \left[r_t\left(-\log \pi_S(y_t \mid s_t; \theta)\right) + (1 - r_t)\mathcal{L}_t^{\text{K-topk}}\right] \tag{12}$$

To provide further flexibility, these two supervision terms can be weighted independently as

$$\mathcal{L}_{\text{CA-OPD}} = \frac{1}{\sum_{t=1}^{L} m_t} \sum_{t=1}^{L} m_t \left[\lambda_{\text{CE}} r_t \mathcal{L}_t^{\text{R}} + \lambda_{\text{KL}} (1 - r_t) \mathcal{L}_t^{\text{K-topk}} \right] \tag{13}$$

4.4 Progressive Transfer of Rollout Control

Early in training, aggressive teacher intervention prevents erratic student transitions from corrupting subsequent states. As the student matures, retaining more student-generated states becomes essential to preserve the on-policy nature of distillation. We therefore progressively relax the confidence requirement.

Let $p_s = s/S$ denote the normalized training progress over S total steps. We employ a cosine schedule to decay the intervention threshold according to

$$\tau_{\text{nll},s} = \tau_{\text{start}} + \frac{\tau_{\text{end}} - \tau_{\text{start}}}{2} \left[1 - \cos(\pi p_s) \right] \tag{14}$$

where $\tau_{\text{start}} < \tau_{\text{end}}$. The lower initial threshold enforces a strict acceptance criterion. As training progresses, the threshold increases, permitting more student proposals to remain. Importantly, this schedule governs the confidence criterion rather than dictating a fixed replacement rate. As the student aligns more closely with the teacher, fewer positions trigger corrections, facilitating a natural transfer of rollout control. The pseudocode for the experimental procedure is provided in Appendix A.

5 Experiments

5.1 Experimental Setup

Models. We employ an SFT-initialized Qwen3.5-0.8B (Qwen Team, 2026) as the student. For supervision, we utilize two separate Qwen3.5-9B models, fine-tuned on their respective domains, to serve as the grounding and OCR teachers. During training, examples are dynamically routed to the corresponding domain teacher.

Baselines. We compare CA-OPD against three representative distillation strategies: *Offline KD*, which trains the student on static teacher-generated trajectories (Hinton et al., 2015; Agarwal et al., 2024); *OPD*, which performs standard on-policy distillation without teacher intervention (Agarwal et al., 2024); and *SKD* (Xu et al., 2025), which uses the teacher's relative ranking of each student proposal to determine whether it is retained in the rollout. We additionally evaluate a CA-OPD variant with a fixed acceptance threshold to isolate the contribution of our progressive schedule. The SFT-initialized student backbone and the teachers are reported as upper and lower reference points rather than controlled baselines.

Evaluation. We evaluate GUI grounding on ScreenSpot-v2 (Wu et al., 2025) and ScreenSpot-Pro (Li et al., 2025), and OCR capabilities on OCRBench-v2 (Fu et al., 2026), CC-OCR (Yang et al., 2025), and OmniDocBench (Ouyang et al., 2025). Furthermore, we evaluate on RefCOCO/+/g (Yu et al., 2016; Nagaraja et al., 2016) and MMBench (Liu et al., 2024) to measure general capability retention. Beyond standard benchmarking, our evaluation includes extensive controlled studies to isolate the impact of individual CA-OPD components. Detailed protocols are provided in Appendix C.

Implementation Details. To ensure fair comparison, all evaluated methods share the exact same student initialization, training data, and optimization budget. Unless otherwise stated, all reported results are averaged over three independent training runs. Detailed SFT and distillation data compositions, hyperparameters, decoding settings, and specific CA-OPD configurations are deferred to Appendix B.

Table 1: **Main results across target and retention benchmarks.** All controlled distillation methods use the same student initialization, training data, and optimization budget, and results are averaged over three independent training runs. For the teacher reference row, ^{G} denotes the grounding teacher and ^{O} denotes the OCR teacher. Best and second-best student results are shown in **bold** and underline, respectively.

Method Method	GUI Grounding		OCR				Retention	
	SS-v2	SS-Pro	OCRv2-EN	OCRv2-ZH	CC-OCR	OmniDoc	RefCOCO	MMBench
Teacher (domain-routed, 9B)	93.88^{G}	62.92^{G}	66.03^{O}	63.87^{O}	78.52^{O}	81.18^{O}	84.56^{G}	90.60^{G}
Qwen3.5-0.8B	79.33	36.42	45.75	43.45	62.19	62.28	78.51	75.21
Student (SFT-init)	80.46	38.19	50.98	51.31	66.40	70.49	**82.00**	79.16
Offline KD (Hinton et al., 2015)	87.96	44.23	51.56	52.91	66.99	70.40	81.86	79.99
OPD (Agarwal et al., 2024)	87.34	40.82	50.79	52.45	63.93	68.57	81.78	80.00
SKD (Xu et al., 2025)	87.63	41.20	51.30	52.43	63.83	68.36	81.81	79.76
CA-OPD, fixed $\alpha = 0.25$	87.81	44.58	52.44	52.76	66.95	**70.64**	81.82	79.73
CA-OPD, annealed $\alpha : 0.50 \rightarrow 0.25$	**88.41**	**45.92**	**52.47**	**53.22**	**67.32**	70.43	81.90	**80.16**

Table 2: **Ablation study of CA-OPD.** We separately examine selective prefix repair and the remaining design choices for rollout control and supervision. *WB* denotes the realized fraction of rollout tokens replaced by teacher predictions. Random intervention uses a comparable writeback budget.

Selective Intervention and Prefix Repair

Variant	Placement	Prefix Repair	SS-Pro	OCRv2-EN	WB (%)
OPD	None	No	40.82	50.79	0.00
Random intervention	Random	Yes	40.20	50.62	3.87
No prefix writeback	Confidence	No	40.92	51.36	0.00
CA-OPD	**Confidence**	**Yes**	**45.92**	**52.47**	5.40

Rollout Control and Supervision

Variant	Schedule	Correction	Loss at Replaced	SS-Pro	OCRv2-EN	WB (%)
Fixed threshold	Fixed	Argmax	CE	44.58	52.44	4.90
Reversed schedule	Reversed	Argmax	CE	44.36	51.69	4.90
Sampled writeback	Annealed	Sample	CE	43.61	51.69	5.60
KL at replaced positions	Annealed	Argmax	KL	45.26	51.75	5.40
CA-OPD	**Annealed**	**Argmax**	**CE**	**45.92**	**52.47**	5.40

5.2 Main Results

Table 1 summarizes the overall results. CA-OPD consistently outperforms standard OPD across all six target benchmarks spanning both GUI grounding and OCR. Crucially, the gains are substantial on complex tasks. For instance, ScreenSpot-Pro surges from 40.82 to 45.92 ($+5.10$ points), and CC-OCR increases from 63.93 to 67.32 ($+3.39$ points). Furthermore, CA-OPD surpasses both SKD and Offline KD across the board. This confirms that our improvements stem directly from the confidence-aware intervention mechanism, rather than the mere presence of on-policy supervision or teacher-generated trajectories.

Validating our dynamic intervention strategy, the annealed variant proves superior to a fixed confidence threshold. It improves five of the six target metrics, which supports the core hypothesis that progressively returning rollout control to the student as it matures is more effective than a static policy. Importantly, these target-task gains do not compromise generalization. When deliberately evaluated on held-out out-of-distribution (OOD) tasks such as RefCOCO and MMBench, CA-OPD maintains performance on par with the SFT initialization and competing baselines, successfully avoiding catastrophic forgetting. Detailed results and an additional evaluation with a Qwen3-VL teacher–student pair are reported in Appendices D and G.1, respectively.

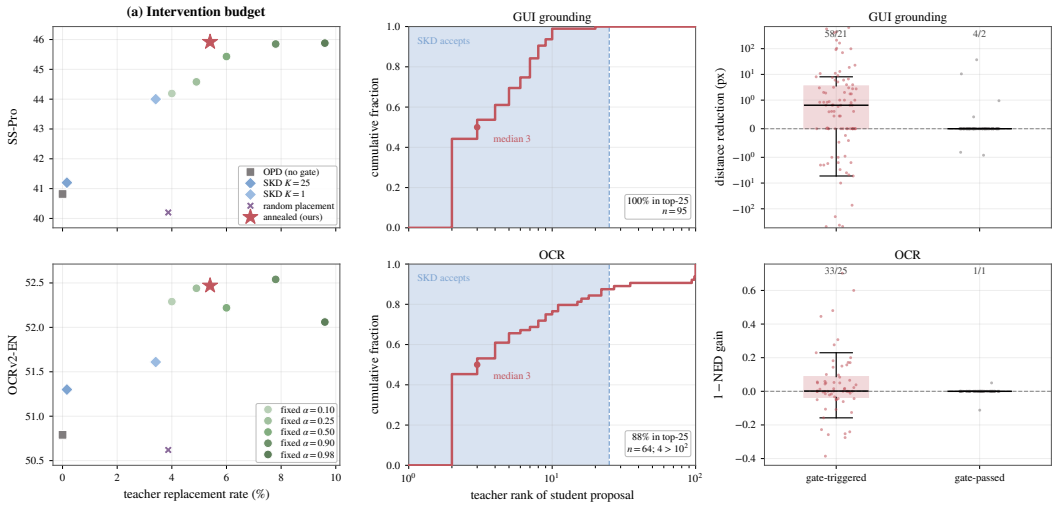

Figure 2: **Analysis of the CA-OPD gating mechanism. (a)** Increasing the intervention budget alone does not recover CA-OPD's gains. **(b)** Most CA-OPD-triggered proposals remain within the teacher's top-25 despite insufficient absolute support. **(c)** Teacher replacement benefits gate-triggered positions, with mostly unchanged gate-passed outcomes; annotations show better/worse counts, excluding ties.

5.3 Ablation Study

We ablate CA-OPD in two stages. We first examine whether the performance gains depend on both explicit prefix repair and intervention placement guided by confidence. We then retain these two core components to isolate the individual effects of rollout scheduling, teacher correction policies, and supervision aligned with intervention decisions. All variants maintain identical student initialization, training data, and optimization budgets.

Selective intervention and prefix repair are both essential. The first block of Table 2 separates the effect of where the teacher intervenes from whether the correction is actually written back into the autoregressive prefix. Randomly inserting teacher corrections at a comparable writeback rate performs similarly to standard OPD. This indicates that merely increasing teacher intervention is insufficient. Conversely, applying confidence gating without prefix writeback also yields performance close to standard OPD, despite preserving the intervention decision and the associated supervision. Only the combination of selective placement and actual prefix repair produces the full improvement. This result demonstrates that CA-OPD benefits heavily from selectively correcting unreliable student transitions and using those corrections to improve the subsequent states visited during training.

Rollout control and supervision provide complementary gains. The second block evaluates the remaining design choices while retaining selective intervention and prefix repair. Replacing the annealed confidence schedule with a fixed threshold degrades performance, and reversing the schedule causes an even larger drop. This finding strongly supports the progressive transfer of rollout control from the teacher to the student. Furthermore, using sampled teacher corrections instead of deterministic argmax writeback reduces performance, suggesting that reliable prefix repair requires a stable correction policy. Finally, replacing direct corrective supervision at replaced positions with KL distillation underperforms the full objective.

5.4 Analysis of the Gating Mechanism

We examine whether CA-OPD's gains arise simply from more frequent interventions, whether absolute teacher support identifies positions missed by rank-based gating, and whether these interventions improve the resulting trajectories. Together, these analyses isolate whether the benefit comes from the intervention budget itself or from selectively correcting particular rollout states.

Intervention frequency alone does not explain the gains. As shown in Figure 2(a), increasing the SKD replacement rate from 0.16% to 3.41% improves SS-Pro from 41.20 to 44.00, but still trails

Table 3: **CA-OPD gains across different data regimes.** Each cell reports the improvement of annealed CA-OPD over OPD. The gains remain positive across all strata rather than being confined to a particular output length, initial competence level, or teacher-rejection density.

Stratification	Low / Short	Medium	High / Long
Output length	Short $+1.50$	Medium $+0.61$	Long $+1.16$
Initial competence	All wrong $+3.90$	Mixed $+10.60$	All correct $+3.65$
Teacher-rejection density	Low ρ $+4.95$	Medium ρ $+5.31$	High ρ $+5.05$

CA-OPD by 1.92 points on SS-Pro and 0.86 points on OCRv2-EN. CA-OPD performance is also non-monotonic in replacement rate, while random interventions at a comparable budget yield only marginal gains (Table 2). Thus, intervention placement matters beyond frequency alone. In particular, simply allocating more teacher tokens does not reproduce the benefit of selectively intervening at low-support states. The full budget sweep is provided in Appendix E.1.

Absolute support identifies positions missed by rank. Figure 2(b) shows that CA-OPD-triggered proposals have a median teacher rank of 3, with 100% of grounding and 88% of OCR proposals remaining within the teacher's top-25. Thus, rank-based gating would retain most proposals rejected by CA-OPD due to insufficient absolute teacher support.

Gate-triggered positions are consequential. Figure 2(c) compares paired student continuations from the same prefix, retaining the proposed token or replacing it with the teacher argmax. Replacement improves 58 of 95 triggered grounding cases and yields a positive mean effect on OCR, while gate-passed scores mostly remain unchanged. Complete counts, including unchanged outcomes, are reported in Appendix Table 12.

Finally, teacher intervention frequency naturally decreases during training, keeping rollouts predominantly student-driven. This decay indicates that the method does not rely on persistently high levels of teacher correction as optimization proceeds. Detailed training dynamics are provided in Appendix E.3.

5.5 Stratified Performance Analysis

We examine whether the gains of CA-OPD are concentrated in particular output-length, initial-competence, or teacher-rejection regimes.

As shown in Table 3, CA-OPD improves over OPD in every stratum, indicating that its gains are not confined to long outputs, low-competence examples, or states with dense teacher rejection. In particular, the nearly constant margin across rejection-density terciles suggests that the benefit extends beyond examples requiring frequent intervention. Although the mixed-competence group exhibits the largest gain, we treat this as descriptive rather than evidence of a specific intermediate-difficulty effect. CA-OPD also better preserves behavior already acquired during SFT. On examples consistently solved by the initialization, OPD decreases performance from 96.03 to 90.18, whereas CA-OPD retains 93.83 (Appendix Table 13). Detailed scores and stratification protocols are provided in Appendix F and Appendix F.1.

6 Conclusion

We introduced CA-OPD, a confidence-aware on-policy distillation framework for structured visual prediction that uses teacher confidence to selectively repair student-generated trajectories and couples rollout intervention with token-level supervision. Across GUI grounding and OCR, CA-OPD consistently improves over standard OPD and competing distillation methods while preserving held-out capabilities. Controlled analyses further show that its gains depend on selective intervention placement, progressive rollout control, and intervention-aligned supervision, highlighting the importance of jointly considering trajectory construction and knowledge transfer in on-policy distillation.

AI USE STATEMENT

In this work, we used generative AI tools as coding assistants for parts of the experimental implementation. AI-generated code was manually reviewed, modified when necessary, and tested by the authors to ensure correctness and reproducibility. We did not use generative AI tools for developing the research ideas, formulating the problem, designing the proposed method, deriving the algorithm, interpreting experimental results, or making scientific claims. We also did not use generative AI tools to generate experimental conclusions or replace the authors' research judgment. Additionally, generative AI tools were used for limited language refinement during manuscript preparation. All AI-assisted content was carefully reviewed and revised by the authors. We take full responsibility for the final content of this work, including all text, claims, experimental results, and artifacts produced with the aid of generative AI.

REFERENCES

Rishabh Agarwal, Nino Vieillard, Yongchao Zhou, Piotr Stanczyk, Sabela Ramos Garea, Matthieu Geist, and Olivier Bachem. On-policy distillation of language models: Learning from self-generated mistakes. In *International Conference on Learning Representations*, volume 2024, pp. 21246–21263, 2024.

Samy Bengio, Oriol Vinyals, Navdeep Jaitly, and Noam Shazeer. Scheduled sampling for sequence prediction with recurrent neural networks. *Advances in neural information processing systems*, 28, 2015.

Yuxuan Cai, Jiangning Zhang, Haoyang He, Xinwei He, Ao Tong, Zhenye Gan, Chengjie Wang, Zhucun Xue, Yong Liu, and Xiang Bai. LLaVA-KD: A framework of distilling multimodal large language models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 239–249, 2025. doi: 10.1109/ICCV51701.2025.00030.

Keqin Chen, Zhao Zhang, Weili Zeng, Richong Zhang, Feng Zhu, and Rui Zhao. Shikra: Unleashing multimodal LLM's referential dialogue magic. *arXiv preprint arXiv:2306.15195*, 2023.

Ting Chen, Saurabh Saxena, Lala Li, David J Fleet, and Geoffrey Hinton. Pix2seq: A language modeling framework for object detection. *arXiv preprint arXiv:2109.10852*, 2021.

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 62nd Annual Meeting of the Association for Computational Linguistics*, pp. 9313–9332, 2024. doi: 10.18653/v1/2024.acl-long.505.

Ling Fu, Zhebin Kuang, Jiajun Song, Mingxin Huang, Biao Yang, Yuzhe Li, Linghao Zhu, Qidi Luo, Xinyu Wang, Hao Lu, et al. Ocrbench v2: An improved benchmark for evaluating large multimodal models on visual text localization and reasoning. *Advances in Neural Information Processing Systems*, 38, 2026.

Yuxian Gu, Li Dong, Furu Wei, and Minlie Huang. Minillm: Knowledge distillation of large language models. In *International Conference on Learning Representations*, volume 2024, pp. 32694–32717, 2024.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

Cheng-Yu Hsieh, Chun-Liang Li, Chih-Kuan Yeh, Hootan Nakhost, Yasuhisa Fujii, Alex Ratner, Ranjay Krishna, Chen-Yu Lee, and Tomas Pfister. Distilling step-by-step! outperforming larger language models with less training data and smaller model sizes. In *Findings of the Association for Computational Linguistics: ACL 2023*, pp. 8003–8017, 2023. doi: 10.18653/v1/2023.findings-acl.507.

Geewook Kim, Teakgyu Hong, Moonbin Yim, Jinyoung Park, Jinyeong Yim, Wonseok Hwang, Sangdoo Yun, Dongyoon Han, and Seunghyun Park. Donut: document understanding transformer without ocr. corr abs/2111.15664 (2021). *arXiv preprint arXiv:2111.15664*, 2021.

Geewook Kim, Teakgyu Hong, Moonbin Yim, JeongYeon Nam, Jinyoung Park, Jinyeong Yim, Wonseok Hwang, Sangdoo Yun, Dongyoon Han, and Seunghyun Park. Ocr-free document understanding transformer. In *European Conference on Computer Vision*, pp. 498–517. Springer, 2022.

Yoon Kim and Alexander M Rush. Sequence-level knowledge distillation. In *Proceedings of the 2016 conference on empirical methods in natural language processing*, pp. 1317–1327, 2016.

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. In *Proceedings of the 33rd ACM International Conference on Multimedia*, pp. 8778–8786, 2025.

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, 2025. doi: 10.1109/CVPR52734.2025.01816.

Yuan Liu, Haodong Duan, Yuanhan Zhang, Bo Li, Songyang Zhang, Wangbo Zhao, Yike Yuan, Jiaqi Wang, Conghui He, Ziwei Liu, et al. Mmbench: Is your multi-modal model an all-around player? In *European conference on computer vision*, pp. 216–233. Springer, 2024.

Jiasen Lu, Christopher Clark, Rowan Zellers, Roozbeh Mottaghi, and Aniruddha Kembhavi. Unified-io: A unified model for vision, language, and multi-modal tasks. *arXiv preprint arXiv:2206.08916*, 2022.

Jiasen Lu, Christopher Clark, Sangho Lee, Zichen Zhang, Savya Khosla, Ryan Marten, Derek Hoiem, and Aniruddha Kembhavi. Unified-io 2: Scaling autoregressive multimodal models with vision, language, audio, and action. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 26429–26445. IEEE, 2024.

Varun K Nagaraja, Vlad I Morariu, and Larry S Davis. Modeling context between objects for referring expression understanding. In *European conference on computer vision*, pp. 792–807. Springer, 2016.

Linke Ouyang, Yuan Qu, Hongbin Zhou, Jiawei Zhu, Rui Zhang, Qunshu Lin, Bin Wang, Zhiyuan Zhao, Man Jiang, Xiaomeng Zhao, et al. Omnidocbench: Benchmarking diverse pdf document parsing with comprehensive annotations. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 24838–24848. IEEE, 2025.

Zhiliang Peng, Wenhui Wang, Li Dong, Yaru Hao, Shaohan Huang, Shuming Ma, and Furu Wei. Kosmos-2: Grounding multimodal large language models to the world. *arXiv preprint arXiv:2306.14824*, 2023.

Qwen Team. Qwen3.5: Towards native multimodal agents, February 2026. URL `https://qwen.ai/blog?id=qwen3.5`.

Marc'Aurelio Ranzato, Sumit Chopra, Michael Auli, and Wojciech Zaremba. Sequence level training with recurrent neural networks. *arXiv preprint arXiv:1511.06732*, 2015.

Stéphane Ross, Geoffrey Gordon, and Drew Bagnell. A reduction of imitation learning and structured prediction to no-regret online learning. In *Proceedings of the fourteenth international conference on artificial intelligence and statistics*, pp. 627–635. JMLR Workshop and Conference Proceedings, 2011.

Yin Wang, Haotian Hu, Jineng Han, Wentao Qiu, Zhenhua Ge, Liujian Tang, and Fanyi Wang. Stepx-edge: An on-device ui vision-language model via architecture-training-deployment co-design. *arXiv preprint arXiv:2607.22708*, 2026.

Zhiyong Wu, Zhenyu Wu, Fangzhi Xu, Yian Wang, Qiushi Sun, Chengyou Jia, Kanzhi Cheng, Zichen Ding, Liheng Chen, Paul Pu Liang, et al. Os-atlas: Foundation action model for generalist gui agents. In *International Conference on Learning Representations*, volume 2025, pp. 5090–5108, 2025.

Wenda Xu, Rujun Han, Zifeng Wang, Long Le, Dhruv Madeka, Lei Li, William Wang, Rishabh Agarwal, Chen-Yu Lee, and Tomas Pfister. Speculative knowledge distillation: Bridging the teacher-student gap through interleaved sampling. In *International Conference on Learning Representations*, volume 2025, pp. 64616–64646, 2025.

Zhibo Yang, Jun Tang, Zhaohai Li, Pengfei Wang, Jianqiang Wan, Humen Zhong, Xuejing Liu, Mingkun Yang, Peng Wang, Shuai Bai, et al. Cc-ocr: A comprehensive and challenging ocr benchmark for evaluating large multimodal models in literacy. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 21744–21754. IEEE, 2025.

Haoxuan You, Haotian Zhang, Zhe Gan, Xianzhi Du, Bowen Zhang, Zirui Wang, Liangliang Cao, Shih-Fu Chang, and Yinfei Yang. Ferret: Refer and ground anything anywhere at any granularity. In *International Conference on Learning Representations*, volume 2024, pp. 57153–57180, 2024.

Licheng Yu, Patrick Poirson, Shan Yang, Alexander C Berg, and Tamara L Berg. Modeling context in referring expressions. In *European conference on computer vision*, pp. 69–85. Springer, 2016.

Qianhao Yuan, Jie Lou, Xing Yu, Hongyu Lin, Le Sun, Xianpei Han, and Yaojie Lu. Vision-opd: Learning to see fine details for multimodal llms via on-policy self-distillation. *arXiv preprint arXiv:2605.18740*, 2026.

A Additional Method and Training Details

Section 4 presents the core CA-OPD formulation. This section provides the surrounding objectives, the exact warm-up adjustment used by the annealing schedule, and a single consolidated training algorithm.

A.1 Objectives and Rollout Context

Supervised fine-tuning conditions on an annotated prefix $y^\star_{<t}$ and minimizes

$$\mathcal{L}_{\mathrm{SFT}} = -\frac{1}{\sum_t m_t} \sum_{t=1}^{L} m_t \log \pi_S(y^\star_t \mid x, y^\star_{<t}; \theta). \tag{15}$$

Offline distillation instead uses teacher-generated prefixes $y^{\mathrm{off}}_{<t}$ and matches the teacher distribution:

$$\mathcal{L}_{\mathrm{off}} = \frac{1}{\sum_t m_t} \sum_{t=1}^{L} m_t D_{\mathrm{KL}}\left(\pi_T(\cdot \mid x, y^{\mathrm{off}}_{<t}) \,\middle\|\, \pi_S(\cdot \mid x, y^{\mathrm{off}}_{<t}; \theta)\right). \tag{16}$$

Thus, SFT and offline KD replay fixed prefixes, standard OPD delegates every transition to the student, and interleaved methods operate between these two endpoints. CA-OPD further couples this mixed state-visitation policy to the supervision rule: teacher-written positions receive cross-entropy, whereas retained student positions receive distributional distillation.

A.2 Warm-Up-Adjusted Annealing Schedule

The threshold schedule in Equation 14 is evaluated using a warm-up-adjusted progress variable. For warm-up fraction ω and total optimizer steps S,

$$p_s = \begin{cases} 0, & s/S \le \omega, \\ \dfrac{s/S - \omega}{1 - \omega}, & s/S > \omega. \end{cases} \tag{17}$$

The acceptance criterion is therefore held at its strictest setting during the first fraction ω of training, after which cosine relaxation begins. The main configuration uses $\omega = 0.1$, $\tau_{\mathrm{start}} = 0.693$, and $\tau_{\mathrm{end}} = 1.386$, corresponding to a teacher-probability floor annealed from $\alpha = 0.50$ to $\alpha = 0.25$ over 471 optimizer steps. This warm-up concerns rollout control only; the optimizer learning-rate schedule has no warm-up.

A.3 Training Algorithm and Chunked Rollouts

Algorithm 1 gives the token-level semantics of rollout construction and intervention-aligned supervision. The algorithm is written token by token for clarity. In the implementation, the student proposes up to 32 tokens per chunk and a teacher query scores the proposed tokens under the shared prefix. If a replacement is triggered, generation restarts from the first replaced position, so every later chunk is conditioned on the updated mixed prefix. These chunked operations preserve the same sequential semantics while reducing query overhead. Their measured cost is reported in Appendix H.

B Implementation Details

All controlled methods in Table 1 use the same student initialization, training data, batch size, and optimizer-step budget. Table 4 collects the resolved configuration of the main annealed CA-OPD run and makes explicit the distinction between optimizer scheduling and gate scheduling.

B.1 Top-k Teacher Transfer

The two teacher truncation settings in Table 4 serve different purposes. During rollout construction, each gate query returns the teacher log-probabilities of the proposed tokens in the current chunk

Algorithm 1 Confidence-Aware On-Policy Distillation for one training example

Require: input x, student π_S, frozen teacher π_T
Require: optimizer step s, total steps S, warm-up fraction ω
Require: τ_{start}, τ_{end}, loss truncation k
$y_{<1} \leftarrow \varnothing$
compute p_s using Equation 17
compute $\tau_{\text{nll},s}$ using Equation 14
for $t = 1$ **to** L **do**
  $s_t \leftarrow (x, y_{<t})$
  sample $\hat{y}_t \sim \pi_S(\cdot \mid s_t; \theta)$
  evaluate $\pi_T(\cdot \mid s_t)$ and $\text{NLL}_{T,t} \leftarrow -\log \pi_T(\hat{y}_t \mid s_t)$
  $r_t \leftarrow \mathbb{I}[\text{NLL}_{T,t} > \tau_{\text{nll},s}]$
  if $r_t = 1$ **then**
    $y_t \leftarrow \arg\max_{v \in \mathcal{V}} \pi_T(v \mid s_t)$
    $\mathcal{L}_t \leftarrow -\log \pi_S(y_t \mid s_t; \theta)$
  else
    $y_t \leftarrow \hat{y}_t$
    compute $\mathcal{L}_t^{\text{K-topk}}$ using Equation 10 and clamp it below at zero
    $\mathcal{L}_t \leftarrow \mathcal{L}_t^{\text{K-topk}}$
  $y_{<t+1} \leftarrow (y_{<t}, y_t)$
return $\left(\sum_{t=1}^{L} m_t \mathcal{L}_t\right) / \left(\sum_{t=1}^{L} m_t\right)$

Table 4: **Training and implementation configuration.** The optimizer settings are shared by every controlled method in Table 1; gate-specific settings apply to CA-OPD variants.

Setting	Value
Models and data	
Student	SFT-initialized Qwen3.5-0.8B
Grounding teacher	Grounding-fine-tuned Qwen3.5-9B
OCR teacher	OCR-fine-tuned Qwen3.5-9B
SFT data	40K OCR examples and 6K RefCOCO examples
Distillation data	15,100 examples: 11,184 grounding and 3,916 OCR
Independent runs	3 for every controlled method in Table 1
Optimization	
Batch size	32
Optimizer steps	471
Learning rate	1×10^{-6}
Optimizer	AdamW, $\beta = (0.9, 0.999)$
Weight decay	0.01
Gradient clipping	1.0
Learning-rate schedule	Constant, with no optimizer warm-up
Precision	bfloat16 mixed precision with fp32 master weights
Rollout and supervision	
Student rollout temperature	1.0
Teacher correction and decoding	Greedy teacher argmax
Gate schedule	Cosine $\alpha : 0.50 \rightarrow 0.25$ after a 0.1 hold fraction
Teacher top k for gate scoring	32
Teacher top k for retained-position loss	64
Maximum prompt length	8,192 tokens
Maximum response length	1,536 tokens
Systems	
Parallelism	FSDP, 4 GPUs for the student
Hardware	$8\times$ NVIDIA A800: 4 student GPUs and 4 teacher replicas

Table 5: **Public datasets used in training.** We use subsets of these datasets for SFT initialization and distillation training.

Dataset	Domain	Link
GUI Grounding		
OS-Atlas	Computer interaction	Hugging Face
RefCOCO	Referring expression grounding	Hugging Face
OCR and Document Understanding		
TextVQA	Scene text understanding	Hugging Face
COCO-Text	Scene text detection / recognition	Hugging Face
DocVQA	Document understanding	Hugging Face
InfographicVQA	Infographic understanding	Hugging Face
VisualMRC	Text-rich image understanding	Hugging Face
Kleister Charity	Key information extraction	Hugging Face
DeepForm	Form understanding	GitHub
ChartQA	Chart understanding	Hugging Face
PubTabNet	Table recognition	Hugging Face
LaTeX-OCR	Formula recognition	Hugging Face
CASIA-HWDB	Handwritten Chinese recognition	Official
CORD	Receipt understanding	Hugging Face
SROIE	Receipt OCR / information extraction	Hugging Face

under the top-32 gate-scoring setting; the gate needs the probability of the exact student proposal and, on rejection, the teacher argmax. At retained positions, the distillation objective uses a forward KL restricted to the teacher's top-64 tokens. The retained probabilities are not renormalized, the resulting per-token value is clamped below at zero, and the retained teacher mass has a training median above 0.999, making the truncation immaterial in practice.

B.2 Training Dataset Sources

We use subsets of the publicly available datasets summarized in Table 5 for SFT initialization and distillation training. Each dataset is processed and filtered according to the requirements of the corresponding task domain.

B.3 Baseline Details

Within each experimental setting, all controlled distillation methods share the same student initialization, teacher configuration, training data, optimization hyperparameters, and optimizer-step budget. All distributional losses use the same teacher top-64 truncation, unnormalized teacher probabilities, and zero-clamping implementation described in Appendix B.1. The methods differ in rollout construction and, for CA-OPD, the supervision assigned to replaced positions.

Offline KD. The teacher generates output sequences before training, and these sequences remain fixed throughout distillation. At each position, the student matches the teacher's next-token distribution conditioned on the same fixed prefix.(Hinton et al., 2015). This baseline therefore uses soft distributional targets rather than hard token targets.

OPD. Following Agarwal et al. (2024), rollouts are generated online by the current student policy. The teacher provides next-token distributions conditioned on the student-generated prefixes, and distributional supervision is applied at every valid response position. No student proposals are replaced during rollout construction.

SKD. We use the rank-based acceptance rule of Xu et al. (2025), retaining a student proposal when its teacher rank is at most K, with $K = 25$ fixed throughout training. Rejected proposals are replaced by the teacher argmax, and generation continues from the resulting mixed prefix. The same top-64 truncated forward-KL loss is applied at all valid response positions, including replaced positions.

C Benchmark and Evaluation Details

Common protocol. Within each benchmark column, every method uses an identical evaluation harness, prompt, preprocessing pipeline, and decoding configuration. Reported means and standard deviations are computed across the three independent training runs; decoding-trial variation within a run is not mixed into the run-level deviations.

GUI grounding. ScreenSpot-v2 and ScreenSpot-Pro use the official Qwen3.5 tool-call prompt, with the assistant turn prefilled up to the coordinate field and images resized by the official smart-resize rule. Each example is decoded eight times using the official sampling configuration: temperature 0.7, top-p 0.8, top-k 20, and presence penalty 1.5. The score is the fraction of trials whose predicted point lies inside the target box.

OCR. OCRBench-v2 is evaluated in English and Chinese with the official scorer. CC-OCR is the macro average over its four tracks, and OmniDocBench v1.5 is reported as full-page $1-\mathrm{NED}$. All OCR benchmarks use greedy decoding with thinking disabled for every model.

Retention. RefCOCO, RefCOCO+, and RefCOCOg comprise eight official splits in total and are evaluated with greedy decoding and ACC@0.5. Table 1 reports their mean over all eight splits, while Appendix D.2 reports each dataset separately. MMBench uses EN-dev-v1.1 accuracy.

D Detailed Results and Run-Level Robustness

This section expands the main results in Table 1 with category-level target scores, dataset-level retention scores, and cross-run standard deviations.

D.1 Fine-Grained Target-Benchmark Results

Table 6: **ScreenSpot-Pro results by application category.** All distillation results are means over three runs. Best and second-best student results are shown in **bold** and underline, respectively; the principal CA-OPD row is shaded.

Method	CAD	Creative	Dev	OS	Office	Scientific	Overall
Student (SFT-init)	29.69	34.42	33.53	40.05	54.62	41.14	38.19
Offline KD	40.01	38.38	38.94	43.75	64.26	44.91	44.23
OPD	36.21	34.54	35.52	37.93	61.99	43.27	40.82
SKD	38.06	34.03	35.72	39.29	60.51	44.49	41.20
CA-OPD, annealed $\alpha : 0.50 \to 0.25$	**41.63**	**39.41**	**41.11**	**45.75**	**65.60**	**47.03**	**45.92**

Table 7: **OCRBench-v2 results by capability category.** English and Chinese scores are macro-averaged where both are available; Referring, Spotting, and Calculation are English-only. All distillation results are means over three runs. Best and second-best student results are shown in **bold** and underline, respectively.

Method	Recog.	Referring	Spotting	Extract.	Parsing	Calc.	Underst.	Reason.
Student (SFT-init)	**70.06**	41.06	18.30	72.27	47.09	**38.03**	54.99	**39.07**
Offline KD	69.02	46.93	22.34	75.48	49.21	36.33	54.46	37.55
OPD	67.95	46.96	19.42	74.30	48.87	36.05	54.51	37.43
SKD	68.16	47.63	22.17	75.66	47.59	35.10	54.28	38.14
CA-OPD, annealed $\alpha : 0.50 \to 0.25$	69.42	**48.32**	**24.63**	**76.16**	**49.30**	36.59	**55.29**	38.00

Tables 6 and 7 show that the gains are not driven by a single subset. CA-OPD is best in all six ScreenSpot-Pro application categories. On OCRBench-v2, it exceeds OPD in all eight capability categories and attains the best listed score in five; the SFT initialization remains strongest in Recognition, Calculation, and Reasoning.

D.2 Held-Out Capability Retention

Table 8 decomposes the aggregate RefCOCO entry in Table 1. Across the controlled distillation methods, the spread is at most 0.43 points in any column, and no method moves a dataset materially away from the SFT initialization. Thus, the target-task gains are not obtained at the expense of the held-out capabilities measured here.

Table 8: **Retention results by dataset.** RefCOCO/+/g are ACC@0.5 averaged over each dataset's official splits; MMBench is EN-dev-v1.1 accuracy. Values are means over three runs. Best and second-best student results are shown in **bold** and underline.

Method	RefCOCO	RefCOCO+	RefCOCOg	MMBench
Student (SFT-init)	**85.63**	**78.08**	82.44	79.16
Offline KD	85.49	77.71	<u>82.62</u>	79.99
OPD	85.38	77.66	82.54	<u>80.00</u>
SKD	85.43	<u>77.77</u>	82.44	79.76
CA-OPD, fixed $\alpha = 0.25$	<u>85.54</u>	77.60	82.55	79.73
CA-OPD, annealed $\alpha : 0.50 \to 0.25$	85.53	<u>77.77</u>	**82.64**	**80.16**

D.3 Cross-Run Variability

Table 9 reports standard deviations across the three independent runs behind each mean in Table 1. Runs share all hyperparameters and differ in data order and stochastic on-policy sampling, so the table captures run-to-run variability rather than decoding-trial variance. Ungated OPD is notably unstable on SS-Pro, CC-OCR, and OmniDoc; the annealed gate lowers those deviations from 1.46, 0.96, and 1.13 to 0.72, 0.51, and 0.33, respectively.

Table 9: **Cross-run standard deviation.** Lower is better. Best and second-best values are shown in **bold** and underline, respectively.

Method	SS-v2	SS-Pro	OCRv2-EN	OCRv2-ZH	CC-OCR	OmniDoc	RefCOCO	MMBench
Offline KD	0.29	**0.40**	0.57	<u>0.17</u>	<u>0.33</u>	**0.07**	<u>0.07</u>	0.18
OPD	0.74	1.46	<u>0.25</u>	0.59	0.96	1.13	0.13	<u>0.06</u>
SKD	<u>0.25</u>	<u>0.41</u>	0.71	0.35	**0.30**	0.48	**0.04**	0.12
CA-OPD, fixed $\alpha = 0.25$	0.47	0.53	**0.17**	0.30	0.38	<u>0.23</u>	0.15	0.20
CA-OPD, annealed $\alpha : 0.50 \to 0.25$	**0.18**	0.72	0.41	**0.10**	0.51	0.33	0.14	**0.05**

E Intervention Budget, Schedule, and Gating Dynamics

E.1 Full Intervention Budget Sweep

Table 10 lists the operating points behind Figure 2(a). The sweep shows that accuracy does not increase monotonically with replacement rate: random placement offers little benefit at a comparable budget, and high-frequency fixed floors do not consistently beat the annealed configuration. This supports the main-text conclusion that the location and timing of interventions matter in addition to their frequency.

E.2 Annealing-Schedule Variants

Threshold selection. We specified $\alpha_{\text{start}} = 0.50$ and $\alpha_{\text{end}} = 0.25$ as the default probability-floor endpoints before conducting the sensitivity experiments. The fixed-threshold ablation uses $\alpha = 0.25$, matching the terminal floor of the main schedule. Subsequent experiments vary the threshold values, schedule shape, and direction to assess sensitivity to these choices.

Table 11 isolates the schedule. Both direction and terminal floor matter: relaxing to $\alpha = 0.10$ loses part of the gain, linear relaxation is weaker than cosine relaxation at the same endpoints, and reversing the schedule loses more. An earlier and stricter start at $\alpha_{\text{start}} = 0.70$ is included as a single-run sensitivity check.

Table 10: **Full intervention-budget sweep.** Replace rate is the median fraction of rollout tokens replaced during training. Values are means over three runs except rows marked †, which are single runs. Bold and underlined values rank only the multi-run configurations.

Variant	Replace rate (%)	SS-Pro	OCRv2-EN
OPD (no gate)	0.00	40.82	50.79
Rank criterion (SKD, $K{=}25$)	0.16	41.20	51.30
Rank criterion, closest budget ($K{=}1$)	3.41	44.00	51.61
Random placement†	3.87	40.20	50.62
Fixed floor $\alpha = 0.10$	4.0	44.19	52.29
Fixed floor $\alpha = 0.25$	4.9	44.58	<u>52.44</u>
Fixed floor $\alpha = 0.50$	6.0	<u>45.43</u>	52.22
Fixed floor $\alpha = 0.90$†	7.8	45.85	52.54
Fixed floor $\alpha = 0.98$†	9.6	45.88	52.06
CA-OPD, annealed $\alpha : 0.50 \to 0.25$	5.4	**45.92**	**52.47**

Table 11: **Annealing-schedule variants.** All rows share the confidence criterion, argmax writeback, and intervention-aligned supervision of the main method. Values are means over three runs except †. Bold and underlined values rank only multi-run configurations.

Schedule	SS-Pro	OCRv2-EN
Cosine $\alpha : 0.50 \to 0.25$ **(main)**	**45.92**	**52.47**
Cosine $\alpha : 0.50 \to 0.10$	<u>45.06</u>	<u>52.27</u>
Linear $\alpha : 0.50 \to 0.10$	44.54	51.61
Reversed cosine $\alpha : 0.10 \to 0.50$	44.61	51.49
Reversed cosine $\alpha : 0.25 \to 0.50$	44.36	51.69
Cosine $\alpha : 0.70 \to 0.10$†	45.37	52.24

E.3 Gating Dynamics over Training

Figure 3 reports the replacement rate throughout training for the annealed and fixed-threshold variants. Under annealing, intervention is more frequent early and decreases as the criterion relaxes. The fixed-threshold run also declines, showing that increasing student reliability independently reduces the number of proposals requiring teacher correction.

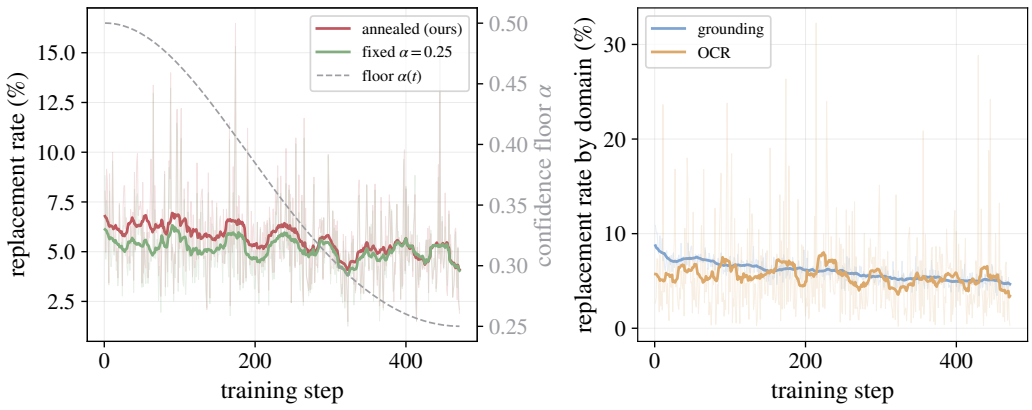

Figure 3: **Gating dynamics over training.** Replacement rate decreases with optimization. For the annealed schedule, the trend reflects both threshold relaxation and increasing student reliability; the fixed-threshold curve isolates the latter effect.

E.4 Counterfactual Outcome Counts

Table 12 reports the complete outcome counts for Figure 2(c). The gate-triggered and gate-passed groups each contain 95 grounding pairs and 64 OCR pairs.

Table 12: **Outcome counts for paired counterfactual comparisons.** Better, Worse, and Unchanged denote positive, negative, and zero differences in the outcome measures plotted in Figure 2(c), respectively. N includes all three outcomes.

Domain	Group	N	Better	Worse	Unchanged	Unchanged (%)
Grounding	Gate-triggered	95	58	21	16	16.8
	Gate-passed	95	4	2	89	93.7
OCR	Gate-triggered	64	33	25	6	9.4
	Gate-passed	64	1	1	62	96.9

F Stratified Protocols and Results

This section supplies the bucket definitions used in Section 5.5 before presenting the corresponding detailed results. All buckets are defined from the SFT initialization before any compared method is trained, so bucket membership is a pre-treatment covariate.

F.1 Stratification Protocols

Output length. OCRBench-v2 examples, with English and Chinese pooled, are sorted by reference transcription length in characters and split into exact tertiles. Each cell in the output-length row of Table 3 reports the within-bucket mean of the official per-example scores, averaged over three training runs. Because task types are pooled, only within-bucket differences between methods are meaningful; the corresponding CA-OPD-over-OPD deltas are therefore summarized directly in the main table rather than repeated as absolute scores here.

Initial competence on SS-Pro. The eight evaluation rollouts of the SFT-initialized student are split in half: the first four define the bucket (all wrong, mixed, or all correct), and every reported score, including the SFT-init row, uses only the remaining four. This split avoids defining and evaluating the buckets on the same trials, which would otherwise force the all-correct initialization score to 100 and inflate selection effects.

Within-type OCR difficulty. OCRBench-v2 uses a single greedy output, so difficulty tertiles are computed within each task type by sorting examples on the SFT-init score. Within-type bucketing is necessary because continuous metrics place partially correct examples in the middle of the range, whereas binary metrics do not; a global split would therefore confound difficulty with metric type and output length.

Teacher-rejection density. For each SS-Pro example, the SFT-init greedy output is teacher-force scored by the grounding teacher. We define ρ as the fraction of output tokens whose teacher probability is below 0.25, the terminal floor of the main annealed schedule, and split examples into tertiles. Thus, low ρ corresponds to strong teacher endorsement and high ρ to frequent rejection. The tercile ordering and the flat CA-OPD-over-OPD margin are unchanged at thresholds 0.1 and 0.5. This statistic correlates with initial competence ($r = -0.51$) but not with output length ($r = -0.10$).

F.2 SS-Pro by Initial Student Competence

Table 13 shows that CA-OPD improves on OPD in every initial-competence group. It also better preserves behavior already mastered by the SFT initialization: in the all-correct group, OPD falls from 96.03 to 90.18, whereas annealed CA-OPD retains 93.83.

Table 13: **SS-Pro performance by initial student competence.** Examples are grouped using four held-out SFT-init rollouts and evaluated using the remaining four. Bold and underlined values denote the best and second-best controlled distillation results; SFT-init is a reference row.

Method	All wrong (n=822)	Mixed (n=312)	All correct (n=447)
Student (SFT-init)	2.49	49.44	96.03
OPD	9.88	51.47	90.18
CA-OPD, fixed $\alpha = 0.25$	<u>13.39</u>	<u>58.60</u>	<u>92.04</u>
CA-OPD, annealed $\alpha : 0.50 \to 0.25$	**13.79**	**62.07**	**93.83**

F.3 OCRBench-v2 by Within-Type Difficulty

Table 14 reports bucket-level micro means. The annealed method improves over OPD in every stratum of both languages; the fixed floor falls below OPD only in the hardest Chinese tertile. The extreme SFT-init values are a consequence of defining the buckets on that same greedy output and should be read as a reference rather than a controlled comparison.

Table 14: **OCRBench-v2 by within-type difficulty tertiles.** Values are bucket-level micro means and are not directly comparable with the macro-averaged headline scores. Bold and underlined values denote the best and second-best controlled distillation results; SFT-init is a reference row.

	OCRv2-EN			OCRv2-ZH		
Method	Hard	Medium	Easy	Hard	Medium	Easy
Student (SFT-init)	18.04	59.74	90.52	15.91	45.05	78.86
OPD	26.26	58.34	81.65	<u>25.82</u>	45.39	72.43
CA-OPD, fixed $\alpha = 0.25$	<u>26.63</u>	<u>59.92</u>	<u>82.70</u>	25.49	<u>45.90</u>	<u>73.36</u>
CA-OPD, annealed $\alpha : 0.50 \to 0.25$	**27.00**	**60.10**	**82.77**	**26.26**	**46.32**	**73.60**

F.4 SS-Pro by Teacher-Rejection Density

Table 15 shows an almost constant annealed-CA-OPD margin over OPD across rejection-density tertiles: $+4.95$, $+5.31$, and $+5.05$. In the low-ρ group, where the initialization is most strongly endorsed by the teacher, OPD falls below the initialization while both gated variants improve on it.

Table 15: **SS-Pro by teacher-rejection-density tertiles.** ρ is the fraction of SFT-init output tokens with teacher probability below 0.25; each tertile contains 527 examples. Scores average eight decoding trials and three training runs. Bold and underlined values denote the best and second-best controlled distillation results; SFT-init is a reference row.

Method	Low ρ	Medium ρ	High ρ
Student (SFT-init)	65.37	37.83	11.36
OPD	63.87	40.23	18.35
CA-OPD, fixed $\alpha = 0.25$	<u>67.61</u>	<u>44.25</u>	<u>21.88</u>
CA-OPD, annealed $\alpha : 0.50 \to 0.25$	**68.82**	**45.53**	**23.40**
Δ (annealed $-$ OPD)	$+4.95$	$+5.31$	$+5.05$

G Additional Comparisons

G.1 Additional Evaluation with Qwen3-VL

We conduct an additional evaluation using Qwen3-VL-8B-Instruct as the frozen teacher and Qwen3-VL-2B-Instruct as the student. Both models are initialized directly from their released Instruct checkpoints, without additional task-specific supervised fine-tuning before distillation. All compared methods use the same Qwen3-VL-2B-Instruct initialization, distillation data, optimization budget,

Table 16: **Additional evaluation with Qwen3-VL.** Results for the distillation methods are reported as mean ± standard deviation over three independent runs, while the undistilled student and teacher are single-checkpoint reference points. Best and second-best results among all student-side configurations, including the undistilled student, are shown in **bold** and underline, respectively, the teacher is excluded from ranking.

	GUI Grounding		OCR				Retention	
Method	SS-v2	SS-Pro	OCRv2-EN	OCRv2-ZH	CC-OCR	OmniDoc	RefCOCO	MMBench
Teacher (Qwen3-VL-8B-Instruct)	94.94	56.20	64.57	62.36	76.38	68.66	88.33	89.60
Student (Qwen3-VL-2B-Instruct)	88.77	41.14	55.61	54.42	64.61	64.90	**84.30**	84.06
Offline KD (Hinton et al., 2015)	91.08 ±0.20	**48.64** ±0.47	57.17 ±0.38	54.78 ±0.27	**66.16** ±0.29	65.95 ±0.59	84.00 ±0.14	84.05 ±0.20
OPD (Agarwal et al., 2024)	90.86 ±0.50	46.40 ±0.85	56.77 ±0.28	54.43 ±0.40	64.89 ±0.07	65.59 ±0.67	84.05 ±0.15	84.00 ±0.13
SKD (Xu et al., 2025)	90.93 ±0.22	46.58 ±0.78	56.65 ±0.32	**54.82** ±0.54	65.01 ±0.55	66.28 ±2.33	84.04 ±0.18	84.08 ±0.00
CA-OPD, annealed $\alpha : 0.50 \to 0.25$	**91.35** ±0.27	48.36 ±0.50	**57.18** ±0.31	54.32 ±0.41	65.87 ±0.57	**67.01** ±2.08	83.82 ±0.14	**84.27** ±0.12

and benchmark evaluation protocols. The training data composition, decoding configurations, and evaluation metrics are aligned with those used in the main Qwen3.5 experiments.

It achieves the strongest results among the compared distillation methods on ScreenSpot-v2, OCRBench-v2 English, OmniDocBench, and MMBench. These results indicate that the benefits of CA-OPD are not restricted to the SFT-initialized Qwen3.5 setting.

G.2 Single-Teacher Grounding Comparison

Table 17 checks that the benefit of confidence gating does not depend on multi-teacher routing or on the SFT initialization used in the main experiments. Here, the student is the raw pretrained Qwen3.5-0.8B, a single grounding teacher supervises one domain, and the gate uses a fixed floor $\alpha = 0.25$ for 699 optimizer steps. Because this is a single-run setting with a different initialization and training regime, its absolute values are not comparable to Table 1; only the within-table OPD–CA-OPD comparison is meaningful. CA-OPD improves SS-v2 by 1.74 points and SS-Pro by 5.48 points based on the displayed values.

Table 17: **Single-teacher grounding comparison.** The student is raw Qwen3.5-0.8B, the gate uses fixed $\alpha = 0.25$, and results come from one run. Grounding follows the main-table protocol of eight sampled trials.

Method	SS-v2	SS-Pro
OPD	87.67	40.64
CA-OPD, fixed $\alpha = 0.25$	**89.41**	**46.12**

G.3 Policy-Gradient Baseline

For completeness, we also trained a policy-gradient variant using a reverse-KL objective with a REINFORCE-style estimator under the same initialization, data, and optimizer budget as the main comparison. In one run it reaches 88.61 on SS-v2, 45.56 on SS-Pro, 51.58 on OCRv2-EN, and 53.04 on OCRv2-ZH. Its SS-Pro score is close to CA-OPD, but it optimizes a different objective from the teacher-distribution matching studied in the main paper and was not repeated across runs. We therefore keep it outside the controlled main table.

H Training Cost and Compute Matching

All methods run on the same 8×A800 node with identical data, batch size, and 471 optimizer steps. Table 18 reports median wall-clock seconds per optimizer step over a full run. Both CA-OPD and SKD cost about 1.5× ungated OPD because they perform interleaved teacher queries during generation; the gate arithmetic itself is not the dominant overhead.

The fixed-α and annealed-α throughput is taken from the run using the final chunked gate implementation. Two earlier runs of that configuration used a per-token implementation at approximately 109

Table 18: **Measured training cost.** Values are median seconds per optimizer step under matched hardware, data, and step count. Offline KD uses fixed teacher prefixes and therefore has no rollout replacement rate.

Method	Seconds/step	Relative to OPD	Median replace rate (%)
Offline KD	35.3	0.89	–
OPD	39.6	1.00	0.00
SKD (K=25)	58.6	1.48	0.16
CA-OPD, fixed $\alpha = 0.25$	58.3	1.47	4.9
CA-OPD, annealed $\alpha : 0.50 \rightarrow 0.25$	59.3	1.50	5.4

seconds per step; this changes throughput, not the reported final scores, whose SS-Pro cross-run deviation is 0.53. The reported timings reflect our current chunked implementation; further systems-level optimization of interleaved rollouts is left for future work.

I Limitations and Scope

CA-OPD assumes that teacher probabilities are sufficiently calibrated to identify unreliable student transitions; systematic teacher–student mismatch, such as domain shift, may require retuning the confidence floor. We validate structured visual prediction with shared teacher–student tokenizers, leaving mismatched tokenizers and open-ended generation untested.

Mixed-prefix rollout remains sequential in chunks and queries the teacher at every visited state, producing about $1.5\times$ the wall-clock cost of ungated OPD (Appendix H). Gating also reduces rather than eliminates erosion of mastered behavior: on consistently solved SS-Pro examples, annealed CA-OPD reaches 93.83 versus the initialization's 96.03.