TRAIN4MERGE: A CONTROLLED SINGLE-TEACHER STUDY OF RL VS. SFT TEACHERS FOR OPD-BASED MODEL MERGING

Anonymous authors
Paper under double-blind review

ABSTRACT

Domain experts trained from a shared checkpoint can be merged into one model through on-policy distillation (OPD), where they act as teachers supervising a student on its own trajectories. One upstream choice is rarely examined: whether to build each expert with supervised fine-tuning (SFT) or reinforcement learning (RL). Yet equally strong teachers need not be equally good teachers. We probe this choice through controlled single-teacher OPD, a building block of multi-teacher OPD: in Agentic, Reasoning, and Perception, comparably performing SFT and RL teachers are trained from Qwen3.5-9B, each guiding a student initialized from it. At their best checkpoints, RL-guided students outperform SFT-guided students by 4.27, 1.50, and 0.86 percentage points in Agentic, Reasoning, and Perception, respectively, and recover more of their teachers' performance gains over the base model. The contrast is clearest in Agentic, where the best SFT-guided student recovers only 44.44% of its teacher's gain, whereas the best RL-guided student recovers 115.00%, surpassing its teacher. Our analysis points to an explanation: RL teachers stay much closer to the shared initialization in parameter space than SFT teachers and are therefore easier for their students to follow.

1 INTRODUCTION

A growing line of work builds a single model with capabilities across many domains in two stages: train a teacher for each domain from a shared checkpoint, then merge their capabilities into one student through multi-teacher OPD (Ma et al., 2026; Yan et al., 2026). The first stage carries a less visible design decision: how to train the teachers. Existing pipelines differ here, some using RL (Ma et al., 2026) and others SFT (Yan et al., 2026; Lian et al., 2026), and the resulting teachers are naturally compared by their own task performance. Yet in OPD, the student does not inherit a teacher's score; it learns from the teacher's token-level supervision on its own trajectories (Agarwal et al., 2024), so teachers with equal scores need not teach equally well. This motivates our central question: **which post-training algorithm produces teachers better suited to OPD when training starts from the same checkpoint—SFT or RL?**

In multi-teacher OPD, teachers sharing one student can transfer abilities across domains or interfere with one another (Men et al., 2026), which could blur the comparison between SFT and RL teachers. For a cleaner comparison, we study single-teacher OPD, the building block of multi-teacher OPD. Because prompts are typically routed to their domain teachers, each teacher supervises the shared student through essentially the single-teacher OPD objective (Ma et al., 2026; Yan et al., 2026), so a teacher that is hard to learn from on its own is likely to remain so alongside others. We contrast comparably performing SFT and RL teachers through controlled single-teacher, single-student OPD in Agentic, Reasoning, and Perception, initializing all models from Qwen3.5-9B (Team, 2026).

The same answer emerges in all three domains: at their best checkpoints, RL-guided students outperform SFT-guided students by 4.27, 1.50, and 0.86 percentage points in Agentic, Reasoning, and Perception, respectively, and recover more of their teachers' performance gains over the base model. In Agentic, this 4.27-point student gap arises from teachers just 0.57 points apart in success rate: the best SFT-guided student recovers only 44.44% of its teacher's gain, whereas the best RL-guided student reaches 115.00% and surpasses its teacher. In Reasoning and Perception, the best RL-guided

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.

students lead even though their teachers score below the SFT teachers on the metric used to pair them.

Why would comparably capable teachers guide their students so differently? Our analysis points to how far each teacher has moved from the shared initialization. RL teachers reach comparable task performance with much smaller parameter displacements than SFT teachers, whereas OPD moves both students by similar amounts early in training. Teacher–student agreement in Top-K predictions also favors RL-guided students: in Agentic and Perception, it rises faster early in OPD despite a slightly lower start, and in Reasoning, it is high from initialization. These observations support the hypothesis that, because OPD adapts the student incrementally, a teacher that departs less from the shared initialization is easier for the student to follow. Perception, whose SFT targets differ only in click coordinates, also fits this hypothesis: its SFT teacher departs less from the initialization than the other domains' RL teachers, and its student gap is the smallest.

A teacher's own score is thus an incomplete guide to its value for OPD. When constructing domain teachers from a shared base model for subsequent OPD, our findings support prioritizing RL. Our main contributions are as follows.

1. **Demonstrating the distillation advantage of RL teachers through controlled single-teacher comparisons.** Under shared initialization and comparable teacher performance, the best RL-guided OPD student in each domain outperforms the best SFT-guided student by 0.86–4.27 percentage points and achieves higher teacher gain recovery.
2. **Providing mechanistic evidence for the RL teacher advantage.** Parameter displacement and Top-K alignment dynamics support the hypothesis that RL teachers' smaller departures from the shared initialization facilitate student learning through OPD.

2 Related Work

Capability integration through model merging and OPD. Model merging seeks to integrate specialized capabilities into a single model. Parameter-space approaches combine task vectors relative to a shared base model (Ilharco et al., 2023), resolve interference between updates (Yadav et al., 2023), or sparsify and rescale task-specific changes before merging (Yu et al., 2024). OPD provides another route by transferring teacher capabilities through supervision on student-generated trajectories (Agarwal et al., 2024), with Qwen3 demonstrating its use for strong-to-weak transfer following off-policy distillation (Yang et al., 2025). Recent work further improves this transfer process through error-aware supervision allocation (Yan et al., 2026) and curricula over interaction depth (Wang et al., 2026). Other approaches use verifiable correctness feedback to filter, gate, or reweight teacher guidance on student-generated trajectories (Huang et al., 2026; Akhondzadeh et al., 2026; Lin et al., 2026; Gan et al., 2026). These advances focus primarily on how teacher guidance is selected and applied during OPD. The upstream choice of how to train teachers for effective capability merging has received less systematic attention.

Expert construction for subsequent OPD. This upstream choice is already present in pipelines that train domain experts before merging their capabilities through OPD, yet existing pipelines adopt different training algorithms. Xiaomi's MOPD trains domain experts through RL from a common general SFT checkpoint and initializes the OPD student from that same checkpoint (Ma et al., 2026). MAGA and UI-MOPD instead use SFT to construct platform-specific teachers from a shared checkpoint within each pipeline (Yan et al., 2026; Lian et al., 2026). Other pipelines combine these algorithms or use different recipes across domains: DeepSeek-V4 applies SFT followed by GRPO to construct specialists for OPD consolidation (DeepSeek-AI et al., 2026), while MiMo-V2-Flash uses both SFT- and RL-trained teachers (Team et al., 2026). Although these pipelines demonstrate several ways to construct experts, they do not establish which training algorithm produces more effective OPD teachers under comparable conditions. This leaves open a central question: when teachers start from the same checkpoint and reach similar task performance, does SFT or RL better support subsequent capability merging?

Teacher compatibility and controlled comparison. Mechanistic studies further motivate examining teacher training as a factor in OPD effectiveness. Rethinking OPD relates transfer effectiveness to teacher–student compatibility and the progressive alignment of high-probability tokens (Li et al.,

2026a). Studies of parameter dynamics observe small student displacements during multi-teacher OPD (Men et al., 2026) and concentrated updates with early subspace alignment (Cai et al., 2026a); Rethinking OPD II further examines how quickly students absorb teacher supervision (Fu et al., 2026). Together, these findings suggest that effective transfer depends on how readily students can acquire teacher capabilities, motivating attention to how those teachers are trained. Our work addresses this upstream choice through a controlled comparison: SFT and RL are applied directly to the same initial checkpoint, the resulting teachers have comparable task performance, and both OPD students are initialized from that same checkpoint. Across Agentic, Reasoning, and Perception, our single-teacher experiments examine which training algorithm better prepares domain experts for subsequent capability merging, complementing work that improves the OPD procedure itself.

3 Experimental Setup

To examine how the teacher-training algorithm affects subsequent OPD, we compare SFT and RL teachers with comparable task performance, using Qwen3.5-9B as the shared initialization for both teachers and students. Our experiments span Agentic, Reasoning, and Perception, with a consistent distillation procedure for the two teacher types within each domain. Teacher and student training hyperparameters and OPD implementation details are provided in Appendix B.

3.1 Teacher Construction and Single-Teacher Distillation

Let the initial Qwen3.5-9B model be $\pi_0 = \pi_{\theta_0}$, and let $\mathcal{D}_d$ denote the training task pool for domain d. In each domain, we start SFT and RL directly from θ_0 to construct the two types of domain teachers. SFT uses task demonstrations for supervised learning, whereas RL employs group relative policy optimization (GRPO) (Shao et al., 2024) to optimize model-generated responses or actions using task rewards.

After teacher training, we select SFT and RL teachers with comparable performance from the saved checkpoints using the domain-specific criteria described in Section 3.3. We denote them by $T^{(d)}_{\mathrm{SFT}}$ and $T^{(d)}_{\mathrm{RL}}$, respectively. Performance matching reduces the influence of differences in the teachers' own task capabilities on the subsequent distillation comparison. Teacher training settings and checkpoint selection criteria are provided in Appendix B.

Both groups of students are then reinitialized from the initial Qwen3.5-9B checkpoint and undergo single-teacher OPD with their respective teachers:

$$S^{(d)}_{\leftarrow a} = \mathrm{OPD}\Big(\pi_0, T^{(d)}_a, \mathcal{D}_d; \mathcal{C}_d\Big), \qquad a \in \{\mathrm{SFT}, \mathrm{RL}\}. \tag{1}$$

Here, $S^{(d)}_{\leftarrow a}$ denotes the student model in domain d obtained through OPD under the supervision of teacher $T^{(d)}_a$, and $\mathcal{C}_d$ denotes the shared distillation configuration within that domain. Distillation uses the training task pool from the corresponding domain. Each current student generates output sequences from the training inputs, and its fixed teacher provides token-level supervision on the student-generated prefixes. The two groups of students use the same training input data and the same OPD configuration, with the teacher-training algorithm as the variable under comparison.

3.2 Training Data

In each domain, the SFT and RL teachers and their corresponding OPD students are trained on the same training set.

Agentic. We use the publicly released mobile training data from UI-MOPD (Lian et al., 2026)[1] as the shared task pool for MobileWorld teacher training. SFT learns from demonstration actions, whereas RL optimizes model-generated actions using task rewards.

Reasoning. We use Geometry3K (Geo3K) (Lu et al., 2021) as the common source of mathematical training data. Both teachers use the Geo3K training problems in the Geometry3K subset of

[1] https://github.com/EliSpectre/UI-MOPD/tree/main/data

NVIDIA's Nemotron-Image-Training-v3 (NVIDIA, 2026): the SFT teacher uses NVIDIA's publicly released complete chain-of-thought annotations as supervision targets, whereas the RL teacher uses the original Geo3K answer labels as reward references.

Perception. We use the publicly released GUI grounding training data from GUI-SD (Zhang et al., 2026)[2] for both teacher-training methods. Each example consists of an interface screenshot, a target description, and a location annotation. SFT learns from demonstration locations, whereas RL optimizes predicted locations using task rewards.

The dataset versions, sizes, and preprocessing procedures for teacher training and OPD inputs are provided in Appendix B.

3.3 Evaluation Benchmarks and Metrics

We evaluate mobile task execution, visual mathematical reasoning, and interface element grounding using the benchmarks and task metrics summarized in Table 1.

Table 1: Evaluation benchmarks and metrics.

Domain	Evaluation benchmarks	Task metric
Agentic	MobileWorld (Kong et al., 2025)	Task success rate
Reasoning	Geo3K test set (Lu et al., 2021), MathVista (Lu et al., 2024), MathVision (Wang et al., 2024), MathVerse (Zhang et al., 2024)	Answer accuracy
Perception	ScreenSpot-v2 (Wu et al., 2024), ScreenSpot-Pro (Li et al., 2025), UI-Vision (Nayak et al., 2025), MMBench-GUI L2 (Wang et al., 2025), OSWorld-G, OSWorld-G-Refine (Xie et al., 2025)	Grounding accuracy

For Agentic evaluation, we use the GUI-only task set of MobileWorld, with a denominator of 117 tasks per round according to the task registry in the code version used. Each model is evaluated over three rounds. The reported mean success rate is the total number of successes across the three rounds divided by 351.

For Reasoning evaluation, we use the Geo3K test set, the testmini splits of MathVista and MathVision, and the Vision-Only setting of MathVerse testmini. We report answer accuracy on each benchmark.

Perception evaluation uses the six GUI grounding benchmarks in Table 1, with the element grounding task from UI-Vision and the L2 element grounding task from MMBench-GUI (Zhang et al., 2026). We abbreviate OSWorld-G-Refine as OSWorld-G-R in the results.

Evaluation criteria. Teacher performance is matched using MobileWorld mean success rate for Agentic, Geo3K accuracy for Reasoning, and the unweighted mean accuracy across six benchmarks for Perception. Let $J_d(\cdot)$ denote this criterion in domain d, with higher scores indicating better performance. We use the following two complementary metrics to evaluate distillation effectiveness.

Domain task performance. We report J_d for student checkpoints throughout OPD alongside the initial Qwen3.5-9B model and both teachers, allowing a direct comparison of student performance under the two types of teacher supervision. Per-benchmark results for all checkpoints are provided in Appendix C.

Teacher performance gain recovery rate. To measure the extent to which a student recovers the teacher's post-training gains, we divide the student's performance gain from OPD by the corre-

[2] https://huggingface.co/datasets/yankie123/GUI-SD-data

sponding teacher's performance gain from SFT or RL:

$$\text{Recovery}_{d,a} = \frac{J_d\left(S_{\leftarrow a}^{(d)}\right) - J_d(\pi_0)}{J_d\left(T_a^{(d)}\right) - J_d(\pi_0)} \times 100\%, \qquad a \in \{\text{SFT}, \text{RL}\}. \tag{2}$$

This metric measures the fraction of the teacher's performance gain over the shared base model that is recovered by the student. Values above 100% indicate that the student surpasses its teacher. We report recovery for the best checkpoint of each student group under J_d.

The initial model, teachers, and students are evaluated using the same procedure and scoring rules on each benchmark. The specific evaluation splits, decoding settings, and cross-benchmark aggregation procedures are provided in Appendix B.

3.4 Diagnostic Metrics

We characterize OPD learning at two levels: how far teachers and students move from the shared initialization in parameter space, and how closely their token predictions agree.

Parameter displacement. For a teacher with parameters θ_T and a student with parameters θ_t after t OPD steps, we measure displacement from the shared initialization θ_0 as

$$\tau_T = \|\theta_T - \theta_0\|_2, \qquad r_t = \|\theta_t - \theta_0\|_2. \tag{3}$$

Both norms are computed over the same 760 non-MTP parameter tensors; for reference, $\|\theta_0\|_2 = 1796.80$. Smaller values indicate less parameter change from the shared initialization.

Top-K overlap. For a token prefix h, let $S_K(\theta, h)$ denote the set of the K highest-probability next tokens under model parameters θ. Teacher–student Top-K overlap at diagnostic point t is

$$O_K(t) = \mathbb{E}_{h \sim \mathcal{H}_t}\left[\frac{|S_K(\vartheta_t, h) \cap S_K(\theta_T, h)|}{K}\right], \tag{4}$$

where ϑ_t denotes the student weights used at that diagnostic point, and $\mathcal{H}_t$ assigns equal weight to valid token positions in student-generated responses, the prefixes on which the teacher supervises the student during OPD. The overlap measures the fraction of top-ranked candidates shared by teacher and student, with larger values indicating closer output alignment. We use $K = 20$ in our analysis. Measurement details are provided in Appendix A.

4 Experimental Results

We first examine how the teacher-training algorithm affects OPD performance in Agentic, then extend the comparison to Reasoning and Perception. Figures 1–3 show student performance across OPD checkpoints, with both teachers and the base model as references. Teacher gain recovery is reported for the best checkpoint of each student group, and per-benchmark results for all checkpoints are provided in Appendix C.

4.1 Agentic

On MobileWorld (Figure 1), teachers with similar task performance produce markedly different outcomes in single-teacher OPD. The SFT and RL teachers achieve success rates of 22.79% and 23.36%, respectively, separated by just 0.57 percentage points. Even at its best checkpoint, the SFT-guided student reaches a success rate of 19.94% and recovers only 44.44% of the teacher's performance gain over the base model, leaving more than half of that gain unrecovered. The RL-guided student instead reaches 24.22% success and 115.00% teacher gain recovery at its best checkpoint, surpassing the teacher itself. Despite the teachers' similar task performance, the best students differ by 4.27 percentage points after OPD, highlighting the importance of the teacher-training algorithm.

The training trajectories further show that the RL teacher's advantage is not limited to the best checkpoints. Although the SFT-guided student leads at the first evaluated checkpoint, the RL-guided

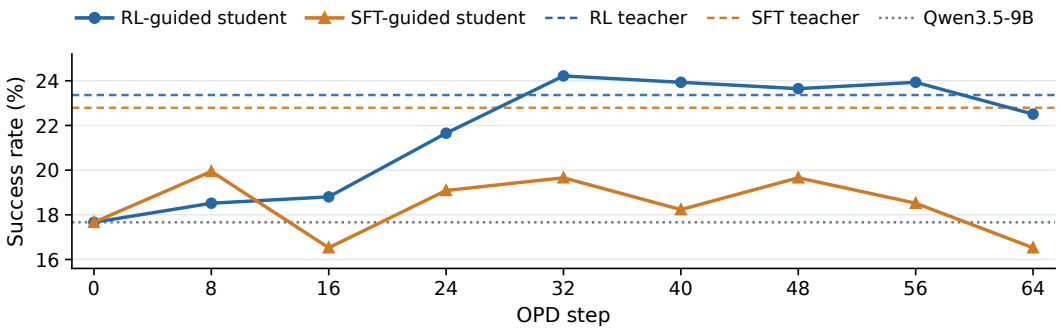


Figure 1: MobileWorld success rates across OPD checkpoints.

student overtakes it by step 16 and leads at all seven checkpoints from steps 16 to 64, with margins of 2.28–5.98 percentage points. The RL-guided student also surpasses its teacher at four of the eight evaluated checkpoints, whereas the SFT-guided student never reaches its teacher's success rate. The RL advantage therefore extends beyond the performance peak and persists across the later evaluated stages of OPD.

Together, these results show that the choice of teacher-training algorithm can substantially affect the effectiveness of OPD. Under shared initialization and comparable teacher task performance, constructing the teacher with RL rather than SFT enables more effective capability transfer, yielding a stronger student and higher teacher gain recovery in our Agentic experiments.

4.2 Reasoning and Perception

Reasoning shows a similar advantage for the RL teacher (Figure 2). On Geo3K, the RL teacher achieves an accuracy of 84.19%, below the SFT teacher's 88.35%, yet its student reaches 91.51% at its best checkpoint, compared with 90.02% for the SFT teacher's student, a gap of 1.50 percentage points. Teacher gain recovery is 262.96% with the RL teacher and 119.23% with the SFT teacher. From step 8 onward, the RL-guided student leads at every checkpoint and remains above both teachers, whereas the SFT-guided student exceeds its teacher only at steps 12 and 16.

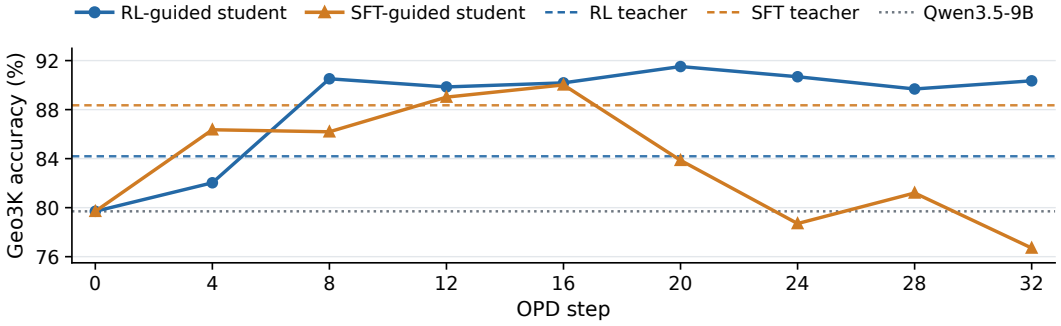


Figure 2: Geo3K accuracies across OPD checkpoints.

Perception exhibits a similar pattern (Figure 3). The RL teacher's mean accuracy across six benchmarks is 67.71%, slightly below the SFT teacher's 68.24%. At their best checkpoints, the corresponding students achieve mean accuracies of 71.22% and 70.36%, with teacher gain recovery of 239.09% and 169.32%, respectively. Both students improve rapidly within the first ten steps. The RL-guided student leads from step 4 to step 20 and reaches its peak earlier, after which the two students perform comparably.

Together, these results support using RL to construct OPD teachers under shared initialization and comparable teacher performance. Our experiments also show that, in Reasoning and Perception, the best students guided by either teacher type recover at least the full teacher performance gain.

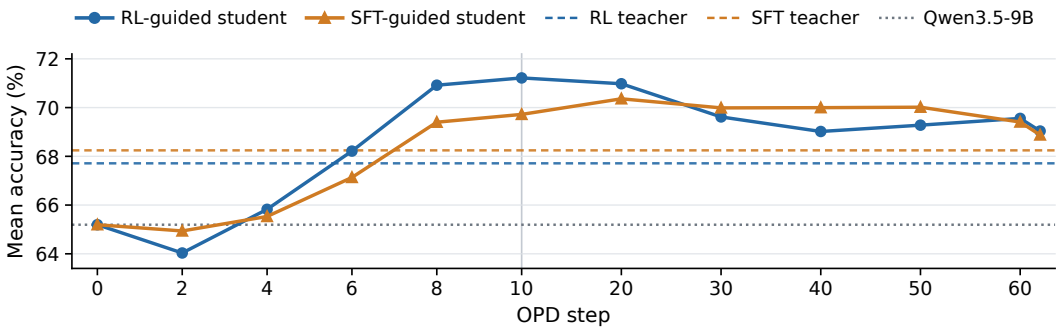


Figure 3: Mean accuracies across the six Perception benchmarks at OPD checkpoints. The horizontal axis is expanded for the first ten steps.

On MobileWorld, however, recovery is only 44.44% with the SFT teacher, compared with 115.00% with the RL teacher. This contrast suggests that the teacher-training algorithm may be particularly important for OPD in Agentic settings such as MobileWorld.

5 Mechanistic Analysis of the RL Teacher Advantage

Why are RL teachers more effective for OPD than SFT teachers with comparable task performance? Prior studies report smaller parameter displacements under RL than under SFT (Yuan et al., 2025; Cai et al., 2026b; Ren et al., 2026). OPD, in turn, adapts the student incrementally: in multi-teacher OPD, students make small updates that keep them close to their initialization (Men et al., 2026); OPD updates concentrate in a low-dimensional subspace that aligns early with the final update direction (Cai et al., 2026a); and students absorb teacher supervision increasingly slowly as training proceeds (Fu et al., 2026). Because a teacher's own parameters reproduce its predictions at distance τ_T from the shared initialization, a smaller τ_T places a teacher-matching solution closer to the student's starting point. We therefore hypothesize that a teacher that departs less from the shared initialization is easier for an incrementally updated student to follow, allowing the student to align with the teacher's predictions more readily and achieve larger task gains. We examine this hypothesis first in parameter space and then in output space.

5.1 Teacher and Student Displacement

Figure 4(a) shows that, in all three domains, the RL teacher departs less from the shared initialization than the SFT teacher: its displacement τ_T is 0.23, 0.19, and 0.37 times that of the SFT teacher in Agentic, Reasoning, and Perception, respectively. The performance-matched RL teachers thus reach comparable task performance with much smaller parameter changes, consistent with prior studies.

OPD, in contrast, moves the two students by similar amounts early in training in all three domains (Appendix A). In Agentic (Figure 4(b)), both students have moved by about 0.38 after 16 steps (0.380 and 0.383), roughly 0.02% of the base-model norm. This movement is already on the scale of the RL teacher's displacement (0.528) but remains far below that of the SFT teacher (2.271); even after 64 steps, the SFT-guided student has moved by only 1.066. Reasoning shows the same contrast: after eight steps, the students have moved by 0.481 and 0.505, compared with teacher displacements of 0.728 and 3.782. In Agentic and Reasoning, early OPD movement is therefore comparable to the RL teacher's displacement but only a small fraction of the SFT teacher's.

In Perception, however, every SFT target is the same click tool call with different coordinates (39 tokens on average, vs. 217 in Agentic and 2,316 in Reasoning; Appendix A). Both of its teachers depart less from the initialization than any teacher in the other domains; even its SFT teacher (0.502) departs less than their RL teachers (0.528 and 0.728). By step 20, the SFT-guided student's r_t (0.726) exceeds its teacher's τ_T, whereas the Agentic and Reasoning SFT-guided students never reach their teachers' τ_T. Consistent with our hypothesis, the RL-guided student's lead is smallest here (0.86 percentage points at the best checkpoints), yet it leads at every checkpoint from step 4 to 20.

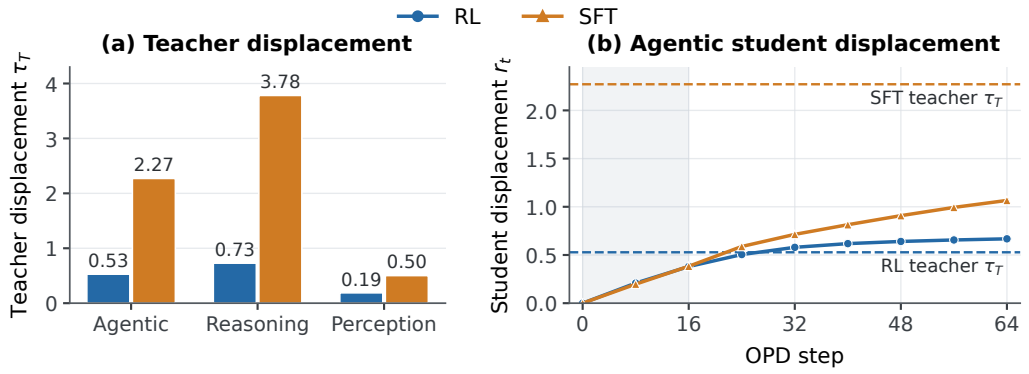


Figure 4: **Teacher and student parameter displacement.** (a) Teacher displacement τ_T. (b) Agentic student displacement r_t; dashed lines mark τ_T. Shading marks the first 16 steps.

5.2 Teacher–Student Prediction Alignment

Because the teacher supervises the student on the student's own prefixes, we measure Top-20 overlap on student rollouts (Figure 5). Rethinking OPD reports that teacher–student Top-K overlap on student-visited states rises steadily in successful OPD runs but stagnates in failing ones (Li et al., 2026a).

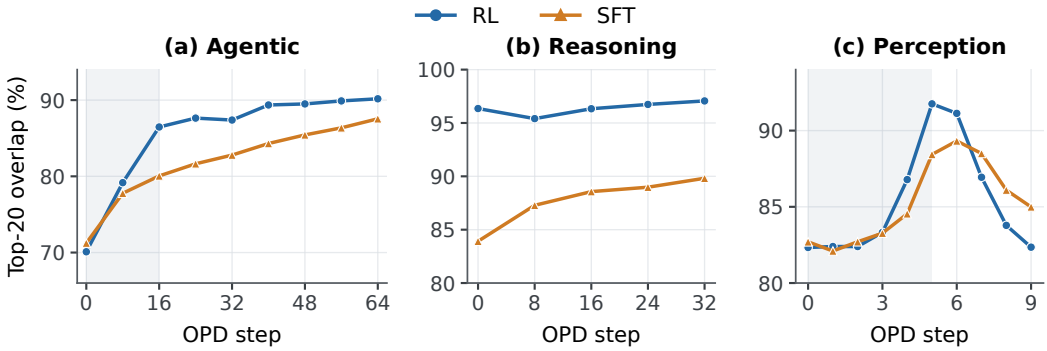


Figure 5: **Teacher–student Top-20 overlap on student rollouts.** Step 0 denotes the base student before its first update. Shading marks the early steps compared in the text.

In Agentic, overlap starts slightly lower in the RL group (70.10% vs. 71.24%), yet rises by 16.37 percentage points (pp) over the first 16 steps, compared with 8.81 pp in the SFT group, and remains higher at every later step. The RL teacher's predictions thus initially differ slightly more from the base model's, yet the student aligns with them faster: this faster alignment does not stem from closer initial output agreement but coincides with the teacher's 4.3-fold smaller parameter displacement. This pattern parallels Rethinking OPD, in which the successful teacher, obtained by RL from the student's own checkpoint, also starts from lower overlap than the failing teacher, yet its overlap rises steadily (Li et al., 2026a).

In Reasoning, the RL-guided student is closely aligned with its teacher from initialization (96.35% vs. 83.91%), and its overlap remains between 95.4% and 97.1%, leaving little room for further growth; meanwhile, its Geo3K accuracy rises by 10.82 points within eight steps. This is consistent with the finding that teachers with additional RL-acquired capability can transfer gains even when overlap is already high (Li et al., 2026a). Overlap in the SFT group rises by 5.9 pp but remains at least 7.2 pp lower at every step.

As in Agentic, overlap in Perception starts slightly lower in the RL group (82.33% vs. 82.70%) yet rises faster: by 9.42 pp over the first five steps, compared with 5.73 pp in the SFT group, and

reaches a higher peak (91.75% vs. 89.32%). This faster early alignment is accompanied by faster improvement: the RL-guided student leads in accuracy from step 4 to step 20 and reaches a higher best score earlier (71.22% at step 10 vs. 70.36% at step 20; Figure 3). Perception thus also supports our hypothesis that a teacher closer to the shared initialization is easier for the student to follow.

Implications for OPD. Prior work associates steadily rising teacher–student overlap with successful distillation and highlights the importance of early alignment (Li et al., 2026a). Consistent with this view, the RL-guided students either align faster early in OPD (Agentic and Perception) or remain closely aligned from the start (Reasoning), and they achieve better task performance than the SFT-guided students (Section 4). Together with the displacement results, these observations support our hypothesis that the RL teacher's smaller departure from the shared initialization makes it easier for OPD to follow, helping the RL-guided student use teacher supervision more effectively.

6 Conclusion

The teacher-to-student transfer underlying OPD-based model merging can depend strongly on how the teacher is trained. Our controlled single-teacher OPD experiments with Qwen3.5-9B show that, under shared initialization and comparable teacher task performance, the best RL-guided students outperform the best SFT-guided students by 4.27, 1.50, and 0.86 percentage points in Agentic, Reasoning, and Perception, respectively, and recover more of their teachers' performance gains over the base model. Further analysis supports a mechanistic hypothesis: the RL teacher's smaller parameter displacement from the shared initialization facilitates student learning through OPD. OPD moves students by similar amounts under either teacher early in training, and RL-guided students align with their teachers' Top-K predictions faster early in OPD (Agentic and Perception) or more closely from the start (Reasoning). These findings identify teacher training as a key design choice for OPD and support prioritizing RL when constructing domain teachers from a shared base model for subsequent OPD. Extending this comparison to multi-teacher OPD is a natural next step (Appendix D).

AI use statement

We used generative AI tools to assist with analyzing experimental data and interpreting results, with providing feedback on the design of our analyses, and with implementing and modifying code for training, evaluation, and plotting. We also used these tools to draft and edit parts of the manuscript for clarity and to search and summarize related literature. We have not used generative AI tools to propose or refine research hypotheses or to generate synthetic data, and this work involves no mathematical claims or proofs. We have reviewed all AI-assisted work: the authors inspected and tested all AI-modified code before use, checked the reported numbers against the underlying experimental records, and reviewed and revised all AI-assisted text. We take full responsibility for the content of this work, including text, claims, and artifacts produced with the aid of generative AI.

Reproducibility statement

All training data and evaluation benchmarks used in this work are publicly available. Section 3 describes teacher construction, single-teacher OPD, the training data, and the evaluation benchmarks and metrics. Appendix B specifies dataset versions, teacher and student training configurations, and checkpoint selection and evaluation. Appendix A details the parameter-displacement and Top-20 overlap measurements, and Appendix C reports per-checkpoint, per-benchmark scores for all students.

References

Rishabh Agarwal, Nino Vieillard, Yongchao Zhou, Piotr Stanczyk, Sabela Ramos, Matthieu Geist, and Olivier Bachem. On-policy distillation of language models: Learning from self-generated mistakes, 2024. URL https://arxiv.org/abs/2306.13649.

Mohammad Sadegh Akhondzadeh, Vijay Lingam, Atula Tejaswi, Chanakya Ekbote, Sujay Sanghavi, and Aleksandar Bojchevski. Reward-gated on-policy distillation, 2026. URL https://arxiv.org/abs/2607.04037.

Yuchen Cai, Ding Cao, Liang Lin, Chunxi Luo, Xin Xu, Kai Yang, Weijie Liu, Saiyong Yang, Tianxiang Zhao, Guangzhong Sun, Guiquan Liu, and Junfeng Fang. Learning to foresee: Unveiling the unlocking efficiency of on-policy distillation, 2026a. URL https://arxiv.org/abs/2605.11739.

Yuchen Cai, Ding Cao, Xin Xu, Zijun Yao, Yuqing Huang, Zhenyu Tan, Benyi Zhang, Guangzhong Sun, Guiquan Liu, and Junfeng Fang. On predictability of reinforcement learning dynamics for large language models, 2026b. URL https://arxiv.org/abs/2510.00553.

DeepSeek-AI, Anyi Xu, Bangcai Lin, Bing Xue, Bingxuan Wang, Bingzheng Xu, Bochao Wu, Bowei Zhang, Chaofan Lin, Chen Dong, Chenchen Ling, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chengyu Hou, Chenhao Xu, Chenze Shao, Chong Ruan, Conner Sun, Damai Dai, Daya Guo, Dejian Yang, Deli Chen, Donghao Li, Dongjie Ji, Erhang Li, Fang Wei, Fangyun Lin, Fangzhou Yuan, Feiyu Xia, Fucong Dai, Guangbo Hao, Guanting Chen, Guoai Cao, Guolai Meng, Guowei Li, Han Yu, Han Zhang, Hanwei Xu, Hao Li, Haofen Liang, Haoling Zhang, Haoming Luo, Haoran Wei, Haotian Yuan, Haowei Zhang, Haowen Luo, Haoyu Chen, Haozhe Ji, Hengqing Zhang, Honghui Ding, Hongxuan Tang, Huanqi Cao, Huazuo Gao, Hui Qu, Hui Zeng, J Yang, JQ Zhu, Jia Luo, Jia Song, Jia Yu, Jialiang Huang, Jialu Cai, Jian Liang, Jiangting Zhou, Jiasheng Ye, Jiashi Li, Jiaxin Xu, Jiewen Hu, Jieyu Yang, Jin Chen, Jin Yan, Jingchang Chen, Jingli Zhou, Jingting Xiang, Jingyang Yuan, Jingyuan Cheng, Jingzi Zhou, Jinhua Zhu, Jiping Yu, Joseph Sun, Jun Ran, Junguang Jiang, Junjie Qiu, Junlong Li, Junmin Zheng, Junxiao Song, Kai Dong, Kaige Gao, Kang Guan, Kexing Zhou, Kezhao Huang, Kuai Yu, Lean Wang, Lecong Zhang, Lei Wang, Leyi Xia, Li Zhang, Liang Zhao, Lihua Guo, Lingxiao Luo, Linwang Ma, Linyan Zhu, Litong Wang, Liyu Cai, Liyue Zhang, Longhao Chen, MS Di, MY Xu, Max Mei, Miaojun Wang, Mingchuan Zhang, Minghua Zhang, Minghui Tang, Mingming Li, Mingxu Zhou, Minmin Han, Ning Wang, Panpan Huang, Panpan Wang, Peixin Cong, Peiyi Wang, Peng Zhang, Qiancheng Wang, Qihao Zhu, Qingyang Li, Qinyu Chen, Qiushi Du, Qiwei Jiang, Rui Tian, Ruifan Xu, Ruijie Lu, Ruiling Xu, Ruiqi Ge, Ruisong Zhang, Ruizhe Pan, Runji Wang, Runqian Chen, Runqiu Yin, Runxin Xu, Ruomeng Shen, Ruoyu Zhang, Ruyi Chen, SH Liu, Shanghao Lu, Shangmian Sun, Shangyan Zhou, Shanhuang Chen, Shaofei Cai, Shaoheng Nie, Shaoqing Wu, Shaoyuan Chen, Shengding Hu, Shengyu Liu, Shiqiang Hu, Shirong Ma, Shiyu Wang, Shuiping Yu, Shunfeng Zhou, Shuting Pan, Shuying Yu, Songyang Zhou, Tao Ni, Tao Yun, Tian Jin, Tian Pei, Tian Ye, Tianle Lin, Tianran Ji, Tianyi Cui, Tianyuan Yue, Tingting Yu, Tun Wang, W Zhang, WL Xiao, Wangding Zeng, Wei An, Weilin Zhao, Wen Liu, Wenfeng Liang, Wenjie Pang, Wenjing Luo, Wenjing Yao, Wenjun Gao, Wenkai Yang, Wenlve Huang, Wenqing Hou, Wentao Zhang, Wenting Ma, Xi Gao, Xiang He, Xiangwen Wang, Xianzu Wang, Xiao Bi, Xiaodong Liu, Xiaohan Wang, Xiaokang Chen, Xiaokang Zhang, Xiaotao Nie, Xiaowen Sun, Xiaoxiang Wang, Xin Cheng, Xin Liu, Xin Xie, Xingchao Liu, Xingchen Liu, Xingkai Yu, Xingyou Li, Xinyu Yang, Xinyu Zhang, Xu Chen, Xuanyu Wang, Xuecheng Su, Xueyin Chen, Xuheng Lin, Xuwei Fu, YC Yan, YQ Wang, YW Ma, Yanfeng Luo, Yang Zhang, Yanhong Xu, Yanru Ma, Yanwen Huang, Yao Li, Yao Li, Yao Xu, Yao Zhao, Yaofeng Sun, Yaohui Wang, Yi Qian, Yi Shao, Yi Yu, Yichao Zhang, Yifan Ding, Yifan Shi, Yijia Wu, Yiliang Xiong, Yiling Ma, Ying He, Ying Tang, Ying Zhou, Yingjia Luo, Yinmin Zhong, Yishi Piao, Yisong Wang, Yixiang Zhang, Yixiao Chen, Yixuan Tan, Yixuan Wei, Yiyang Ma, Yiyuan Liu, Yonglun Yang, Yongqiang Guo, Yongtong Wu, Yu Wu, YuKun Li, Yuan Cheng, Yuan Ou, Yuanfan Xu, Yuanhao Li, Yuduan Wang, Yuehan Yang, Yuer Xu, Yuhan Wu, Yuhao Meng, Yuheng Zou, Yukun Zha, Yunfan Xiong, Yupeng Chen, Yuping Lin, Yuqian Cao, Yuqian Wang, Yushun Zhang, Yuting Yan, Yutong Lin, Yuxian Gu, Yuxiang Luo, Yuxiang You, Yuxuan Liu, Yuxuan Zhou, Yuyang Zhou, Yuzhen Huang, ZF Wu, Zehao Wang, Zehua Zhao, Zehui Ren, Zekai Zhang, Zhangli Sha, Zhe Fu, Zhe Ju, Zhean Xu, Zhenda Xie, Zhengyan Zhang, Zheren Gao, Zhewen Hao, Zhibin Gou, Zhicheng Ma, Zhigang Yan, Zhihong Shao, Zhixian Huang, Zhixuan Chen, Zhiyu Wu, Zhizhou Ren, Zhongyu Wu, Zhuoshu Li, Zhuping Zhang, Zian Xu, Zihao Wang, Zihua Qu, Zihui Gu, Zijia Zhu, Zilin Li, Zipeng Zhang, Ziwei Xie, Ziyi Gao, Ziyi Wan, Zizheng Pan, and Zongqing Yao. Deepseek-v4: Towards highly efficient million-token context intelligence, 2026. URL https://arxiv.org/abs/2606.19348.

Zixuan Fu, Bingxiang He, Yuxin Zuo, Haohuan Huang, Jinqian Zhang, Ruhang Xiao, Cheng Qian, Qinyu Luo, Huan ang Gao, Yudong Wang, Zhiyuan Liu, Ning Ding, and Chaojun Xiao. Rethinking on-policy distillation of large language models ii: One training example, 2026. URL https://arxiv.org/abs/2609.04172.

Siyuan Gan, Yuhan Li, Xiran Wang, Linjian Meng, Boyan Wang, Zhen Zhao, Jing Huo, and Yang Gao. When teacher guidance misleads: Reward-aligned on-policy distillation, 2026. URL https://arxiv.org/abs/2608.27960.

Jingyuan Huang, Zuming Huang, Yucheng Shi, Tianze Yang, Xiaoming Zhai, Wei Chu, and Ninghao Liu. Trust the right teacher: Quality-aware self-distillation for gui grounding, 2026. URL https://arxiv.org/abs/2606.18101.

Gabriel Ilharco, Marco Tulio Ribeiro, Mitchell Wortsman, Suchin Gururangan, Ludwig Schmidt, Hannaneh Hajishirzi, and Ali Farhadi. Editing models with task arithmetic, 2023. URL https://arxiv.org/abs/2212.04089.

Quyu Kong, Xu Zhang, Zhenyu Yang, Nolan Gao, Chen Liu, Panrong Tong, Chenglin Cai, Hanzhang Zhou, Jianan Zhang, Liangyu Chen, Zhidan Liu, Steven Hoi, and Yue Wang. Mobile-world: Benchmarking autonomous mobile agents in agent-user interactive and mcp-augmented environments, 2025. URL https://arxiv.org/abs/2512.19432.

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, 2025. URL https://arxiv.org/abs/2504.07981.

Yaxuan Li, Yuxin Zuo, Bingxiang He, Jinqian Zhang, Chaojun Xiao, Cheng Qian, Tianyu Yu, Huan ang Gao, Wenkai Yang, Zhiyuan Liu, and Ning Ding. Rethinking on-policy distillation of large language models: Phenomenology, mechanism, and recipe, 2026a. URL https://arxiv.org/abs/2604.13016.

Zhongzhi Li, Xuansheng Wu, Yijiang Li, Lijie Hu, and Ninghao Liu. Less is enough: Synthesizing diverse data in llm feature space with sparse autoencoders, 2026b. URL https://arxiv.org/abs/2602.10388.

Zongxia Li, Zhongzhi Li, Yucheng Shi, Ruhan Wang, Junyao Yang, Zhichao Liu, Xiyang Wu, Anhao Li, Yue Yu, Ninghao Liu, Lichao Sun, Haotao Mi, and Leowei Liang. Long-horizon-terminal-bench: Testing the limits of agents on long-horizon terminal tasks with dense reward-based grading, 2026c. URL https://arxiv.org/abs/2607.08964.

Niu Lian, Tongbo Chen, Zhehao Yu, Chengzhen Duan, Fazhan Liu, Hui Liu, Pei Fu, Jian Luan, Heng Qu, Shu-Tao Xia, and Jinpeng Wang. Ui-mopd: Multi-platform on-policy distillation for unified gui agents, 2026. URL https://arxiv.org/abs/2607.04425.

Wenze Lin, Jiale Zhao, Xitai Jiang, Songde Rao, Yining Li, Shenzhi Wang, Bingxiang He, and Gao Huang. On-policy distillation with verifiable reward, 2026. URL https://arxiv.org/abs/2608.24696.

Pan Lu, Ran Gong, Shibiao Jiang, Liang Qiu, Siyuan Huang, Xiaodan Liang, and Song-Chun Zhu. Inter-gps: Interpretable geometry problem solving with formal language and symbolic reasoning, 2021. URL https://arxiv.org/abs/2105.04165.

Pan Lu, Hritik Bansal, Tony Xia, Jiacheng Liu, Chunyuan Li, Hannaneh Hajishirzi, Hao Cheng, Kai-Wei Chang, Michel Galley, and Jianfeng Gao. Mathvista: Evaluating mathematical reasoning of foundation models in visual contexts, 2024. URL https://arxiv.org/abs/2310.02255.

Wenhan Ma, Jianyu Wei, Liang Zhao, Hailin Zhang, Bangjun Xiao, Lei Li, Qibin Yang, Bofei Gao, Yudong Wang, Rang Li, Jinhao Dong, Zhifang Sui, and Fuli Luo. Mopd: Multi-teacher on-policy distillation for capability integration in llm post-training, 2026. URL https://arxiv.org/abs/2606.30406.

Tianyi Men, Zhuoran Jin, Kang Liu, and Jun Zhao. The physics of multi-turn long-horizon planning: From pre-training to post-training via single- and multi-teacher on-policy agentic distillation, 2026. URL https://arxiv.org/abs/2607.24720.

Shravan Nayak, Xiangru Jian, Kevin Qinghong Lin, Juan A. Rodriguez, Montek Kalsi, Rabiul Awal, Nicolas Chapados, M. Tamer Özsu, Aishwarya Agrawal, David Vazquez, Christopher Pal, Perouz Taslakian, Spandana Gella, and Sai Rajeswar. Ui-vision: A desktop-centric gui benchmark for visual perception and interaction, 2025. URL https://arxiv.org/abs/2503.15661.

NVIDIA. Geometry3K Qwen CoT, in Nemotron Image Training v3. Hugging Face dataset, 2026. URL https://huggingface.co/datasets/nvidia/Nemotron-Image-Training-v3/blob/main/geometry3k/README.md.

Zixuan Ren, Jinliang Lu, Junhong Wu, Yang Zhao, Dai Dai, Hua Wu, Haifeng Wang, and Chengqing Zong. Enough is as good as a feast: A comprehensive analysis of how reinforcement learning mitigates task conflicts in llms, 2026. URL https://arxiv.org/abs/2607.22039.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. Deepseekmath: Pushing the limits of mathematical reasoning in open language models, 2024. URL https://arxiv.org/abs/2402.03300.

Guangming Sheng, Chi Zhang, Zilingfeng Ye, Xibin Wu, Wang Zhang, Ru Zhang, Yanghua Peng, Haibin Lin, and Chuan Wu. Hybridflow: A flexible and efficient rlhf framework. In *Proceedings of the Twentieth European Conference on Computer Systems*, EuroSys '25, pp. 1279–1297. ACM, March 2025. doi: 10.1145/3689031.3696075. URL http://dx.doi.org/10.1145/3689031.3696075.

Core Team, Bangjun Xiao, Bingquan Xia, Bo Yang, Bofei Gao, Bowen Shen, Chen Zhang, Chenhong He, Chiheng Lou, Fuli Luo, Gang Wang, Gang Xie, Hailin Zhang, Hanglong Lv, Hanyu Li, Heyu Chen, Hongshen Xu, Houbin Zhang, Huaqiu Liu, Jiangshan Duo, Jianyu Wei, Jiebao Xiao, Jinhao Dong, Jun Shi, Junhao Hu, Kainan Bao, Kang Zhou, Lei Li, Liang Zhao, Linghao Zhang, Peidian Li, Qianli Chen, Shaohui Liu, Shihua Yu, Shijie Cao, Shimao Chen, Shouqiu Yu, Shuo Liu, Tianling Zhou, Weijiang Su, Weikun Wang, Wenhan Ma, Xiangwei Deng, Bohan Mao, Bowen Ye, Can Cai, Chenghua Wang, Chengxuan Zhu, Chong Ma, Chun Chen, Chunan Li, Dawei Zhu, Deshan Xiao, Dong Zhang, Duo Zhang, Fangyue Liu, Feiyu Yang, Fengyuan Shi, Guoan Wang, Hao Tian, Hao Wu, Heng Qu, Hongfei Yi, Hongxu An, Hongyi Guan, Xing Zhang, Yifan Song, Yihan Yan, Yihao Zhao, Yingchun Lai, Yizhao Gao, Yu Cheng, Yuanyuan Tian, Yudong Wang, Zhen Tang, Zhengju Tang, Zhengtao Wen, Zhichao Song, Zhixian Zheng, Zihan Jiang, Jian Wen, Jiarui Sun, Jiawei Li, Jinlong Xue, Jun Xia, Kai Fang, Menghang Zhu, Nuo Chen, Qian Tu, Qihao Zhang, Qiying Wang, Rang Li, Rui Ma, Shaolei Zhang, Shengfan Wang, Shicheng Li, Shuhao Gu, Shuhuai Ren, Sirui Deng, Tao Guo, Tianyang Lu, Weiji Zhuang, Weikang Zhang, Weimin Xiong, Wenshan Huang, Wenyu Yang, Xin Zhang, Xing Yong, Xu Wang, Xueyang Xie, Yilin Jiang, Yixin Yang, Yongzhe He, Yu Tu, Yuanliang Dong, Yuchen Liu, Yue Ma, Yue Yu, Yuxing Xiang, Zhaojun Huang, Zhenru Lin, Zhipeng Xu, Zhiyang Chen, Zhonghua Deng, Zihan Zhang, and Zihao Yue. Mimo-v2-flash technical report, 2026. URL https://arxiv.org/abs/2601.02780.

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

Jiaqi Wang, Wenhao Zhang, Weijie Shi, Yaliang Li, and James Cheng. Tcod: Exploring temporal curriculum in on-policy distillation for multi-turn autonomous agents, 2026. URL https://arxiv.org/abs/2604.24005.

Ke Wang, Junting Pan, Weikang Shi, Zimu Lu, Mingjie Zhan, and Hongsheng Li. Measuring multimodal mathematical reasoning with math-vision dataset, 2024. URL https://arxiv.org/abs/2402.14804.

Xuehui Wang, Zhenyu Wu, JingJing Xie, Zichen Ding, Bowen Yang, Zehao Li, Zhaoyang Liu, Qingyun Li, Xuan Dong, Zhe Chen, Weiyun Wang, Xiangyu Zhao, Jixuan Chen, Haodong Duan, Tianbao Xie, Chenyu Yang, Shiqian Su, Yue Yu, Yuan Huang, Yiqian Liu, Xiao Zhang, Yanting Zhang, Xiangyu Yue, Weijie Su, Xizhou Zhu, Wei Shen, Jifeng Dai, and Wenhai Wang. Mmbench-gui: Hierarchical multi-platform evaluation framework for gui agents, 2025. URL https://arxiv.org/abs/2507.19478.

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: A foundation action model for generalist gui agents, 2024. URL https://arxiv.org/abs/2410.23218.

Tianbao Xie, Jiaqi Deng, Xiaochuan Li, Junlin Yang, Haoyuan Wu, Jixuan Chen, Wenjing Hu, Xinyuan Wang, Yuhui Xu, Zekun Wang, Yiheng Xu, Junli Wang, Doyen Sahoo, Tao Yu, and Caiming Xiong. Scaling computer-use grounding via user interface decomposition and synthesis, 2025. URL https://arxiv.org/abs/2505.13227.

Prateek Yadav, Derek Tam, Leshem Choshen, Colin Raffel, and Mohit Bansal. Ties-merging: Resolving interference when merging models, 2023. URL https://arxiv.org/abs/2306.01708.

Hang Yan, Zhangxuan GU, Beitong Zhou, Jiaxuan Chen, Runze Li, Yusong Hu, Shuheng Shen, and Changhua Meng. Maga: Multi-platform self-fusion of gui agents via structured action distillation, 2026. URL https://arxiv.org/abs/2607.29320.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, Chujie Zheng, Dayiheng Liu, Fan Zhou, Fei Huang, Feng Hu, Hao Ge, Haoran Wei, Huan Lin, Jialong Tang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jing Zhou, Jingren Zhou, Junyang Lin, Kai Dang, Keqin Bao, Kexin Yang, Le Yu, Lianghao Deng, Mei Li, Mingfeng Xue, Mingze Li, Pei Zhang, Peng Wang, Qin Zhu, Rui Men, Ruize Gao, Shixuan Liu, Shuang Luo, Tianhao Li, Tianyi Tang, Wenbiao Yin, Xingzhang Ren, Xinyu Wang, Xinyu Zhang, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yinger Zhang, Yu Wan, Yuqiong Liu, Zekun Wang, Zeyu Cui, Zhenru Zhang, Zhipeng Zhou, and Zihan Qiu. Qwen3 technical report, 2025. URL https://arxiv.org/abs/2505.09388.

Le Yu, Bowen Yu, Haiyang Yu, Fei Huang, and Yongbin Li. Language models are super mario: Absorbing abilities from homologous models as a free lunch, 2024. URL https://arxiv.org/abs/2311.03099.

Xiangchi Yuan, Xiang Chen, Tong Yu, Dachuan Shi, Can Jin, Wenke Lee, and Saayan Mitra. Mitigating forgetting between supervised and reinforcement learning yields stronger reasoners, 2025. URL https://arxiv.org/abs/2510.04454.

Renrui Zhang, Dongzhi Jiang, Yichi Zhang, Haokun Lin, Ziyu Guo, Pengshuo Qiu, Aojun Zhou, Pan Lu, Kai-Wei Chang, Peng Gao, and Hongsheng Li. Mathverse: Does your multi-modal llm truly see the diagrams in visual math problems?, 2024. URL https://arxiv.org/abs/2403.14624.

Yan Zhang, Daiqing Wu, Huawen Shen, Can Ma, and Yu Zhou. Learn where to click from yourself: On-policy self-distillation for gui grounding, 2026. URL https://arxiv.org/abs/2605.00642.

A Additional Details for the Mechanistic Analysis

Parameter displacement. All parameter comparisons use the shared initialization and the same 760 non-MTP tensors, comprising 9,409,813,744 parameters. Displacements use the saved checkpoint weights, with differences and squared sums accumulated in FP64; the base parameter norm is 1796.7960. Relative to this norm, teacher displacements are $0.02938\%/0.12640\%$ for Agentic, $0.04054\%/0.21050\%$ for Reasoning, and $0.01048\%/0.02795\%$ for Perception (RL/SFT). Reasoning teachers are saved in FP32 and the other teachers in BF16; after casting both Reasoning teachers to BF16, the RL teacher's displacement remains below one fifth of the SFT teacher's. Restricted to non-visual tensors and normalized by the base norm of those tensors, the Reasoning teacher displacements are $0.05599\%/0.28381\%$, an RL-to-SFT ratio of 0.20.

Table 2 lists student displacements r_t. At the early-training checkpoints in Table 2 (through step 16 in Agentic and Reasoning and at step 10 in Perception), the RL-to-SFT ratio of r_t lies between 0.88 and 1.06 in all three domains, whereas the corresponding teacher-displacement ratios are 0.23, 0.19, and 0.37. In Perception, this ratio stays between 0.87 and 0.97 through step 62, and the SFT-guided student's r_t exceeds its teacher's τ_T at every checkpoint from step 20; in Agentic and Reasoning, the SFT-guided student's r_t remains below its teacher's τ_T at every checkpoint.

Table 2: Student parameter displacement r_t and teacher displacement τ_T.

Agentic			Reasoning			Perception		
Step	RL	SFT	Step	RL	SFT	Step	RL	SFT
0	0	0	0	0	0	0	0	0
8	0.209	0.197	8	0.481	0.505	10	0.374	0.384
16	0.380	0.383	16	0.715	0.812	20	0.629	0.726
24	0.505	0.588	24	0.828	0.981	30	0.760	0.864
32	0.579	0.714	32	0.901	1.099	40	0.855	0.969
40	0.618	0.814				50	0.942	1.055
48	0.640	0.908				60	1.030	1.119
56	0.656	0.993				62	1.047	1.133
64	0.668	1.066						
τ_T	0.528	2.271	τ_T	0.728	3.782	τ_T	0.188	0.502

Training output formats. Table 3 summarizes the SFT targets, with lengths counted by the Qwen3.5-9B tokenizer over the full target text, including all markup. Every Perception target is a single `left_click` tool call with no reasoning text and differs from the others only in its click coordinates, so masking the coordinates leaves a single distinct target. Perception student rollouts are similarly short from the start: 2,238 of the 2,240 rollouts recorded over OPD steps 0–9, including those of the base model, contain 33–36 response tokens. By contrast, every Agentic target contains a free-form rationale (154 tokens on average) followed by an action description and a tool call, spanning eight action types, and all 25,904 targets remain distinct after masking numbers. Every Reasoning target contains a chain of thought and a boxed final answer, with the chain of thought accounting for 98.1% of all target characters.

Table 3: SFT targets. Lengths are in tokens.

	Agentic	Reasoning	Perception
Number of targets	25,904	2,075	6,990
Mean length	217.02	2,315.89	38.55
Median length	218	1,099	39
10th–90th percentile	181–254	356–6,140	38–39
Free-form reasoning	Rationale	Chain of thought	None

Overlap measurement. We compute the overlap in Equation (4) by assigning equal weight to every valid response-token position; teacher and student are scored at the same causal positions within each group's student-generated prefixes. Step 0 scores both teachers on responses generated by the

base student: 1,024, 32, and 112 responses in Agentic, Reasoning, and Perception, respectively. Later diagnostics use prompts from the corresponding training batches and each group's generated prefixes. Reasoning reports steps 0, 8, 16, 24, and 32, and Perception reports every step from 0 to 9 (Figure 5(c)).

B Experimental Details

B.1 Training Data

Agentic uses 25,904 step-level examples from 2,640 trajectories in the UI-MOPD Uni-GUI-OpenMobile release (revision `774eb20f9772`), retaining its original split and input contexts. Reasoning draws inputs from 2,078 Geo3K training problems aligned with NVIDIA's Nemotron-Image-Training-v3 Geometry3K subset (revision `7656391d4d4c`). Perception uses 6,990 GUI-SD training examples (file SHA-256 prefix `845d6baa50be`). In each domain, both student groups use the same training input pool. OPD uses the original prompts and images without reference answers or additional teacher inputs.

B.2 Teacher Training Configurations

Table 4 summarizes the teacher settings. All teachers start from Qwen3.5-9B and use the training data described above. SFT minimizes token-level cross-entropy on demonstration responses. RL uses GRPO with sequence-level task rewards and no entropy or length terms; rollouts use temperature 1.0 (0.9 in Perception), top-p 1.0, and no top-k truncation. Optimizer betas are $(0.9, 0.999)$, or $(0.9, 0.95)$ for the Reasoning SFT teacher, and the gradient clipping norm is 1. No teacher uses LoRA. The Reasoning RL teacher freezes the visual encoder and merger, whereas the Reasoning SFT teacher updates all parameters.

Table 4: Teacher training settings. Batch: demonstrations (SFT) or prompts $\times$ sampled responses (RL) per update. Max tokens: total (SFT) or prompt/response (RL). Learning rates are configured values.

	Agentic		Reasoning		Perception	
Parameter	SFT	RL	SFT	RL	SFT	RL
Optimizer	AdamW	AdamW	AdamW	AdamW	AdamW	AdamW
Learning rate	10^{-6}	10^{-6}	5×10^{-6}	10^{-6}	2.5×10^{-6}	2.5×10^{-6}
Weight decay	0.01	0	0.01	0.01	0.01	0.01
Batch	32	128×8	32	16×5	112	14×8
Max tokens	16,384	8,192/512	16,384	8,192/8,192	8,320	19,872/128
KL coefficient	–	0	–	0.01	–	0.04

Agentic. SFT targets are built by UI-MOPD's released converter from each demonstration step's rationale, action description, and tool call. The RL teacher uses UI-MOPD's released GUI-action reward: $+1$ if the parsed tool call matches the reference action type and its arguments (e.g., a coordinate inside the annotated element box or case-insensitive text), -0.5 for any other parsable tool call, and -1 if no parsable tool call is produced. Advantages subtract the group mean without standard-deviation normalization, and the clipping ratios are 0.2/0.28. Each generation batch samples eight responses for each of 384 prompts; prompts whose responses include both $+1$ and lower rewards are retained until 128 fill an optimizer batch. Both Agentic teachers follow UI-MOPD (Lian et al., 2026): the SFT teacher uses the optimizer, learning rate, batch size, and maximum sequence length of its released teacher-SFT script, and the RL teacher uses the batch sizes, sequence limits, learning rate, sampling, and clipping settings of its GRPO-based student training, without the distillation loss.

Reasoning and Perception. The Reasoning SFT teacher uses NVIDIA's released chain-of-thought demonstrations (NVIDIA, 2026), whose final answers all match the Geo3K labels; the three demonstrations longer than 16,384 tokens are excluded, leaving 2,075. The Reasoning RL teacher draws its prompts from the same Geo3K training problems, and its reward is 1 if the final

option letter parsed from the response matches the original Geo3K label and 0 otherwise, including responses without a parsable final answer. Each iteration samples five responses for each of 64 prompts and performs four updates of 16 prompts; the learning rate, batch sizes, sample count, and KL setting follow the Qwen3.5 Geo3K GRPO example in verl (Sheng et al., 2025). Perception SFT targets a click at the center of the ground-truth box in 0–1000 coordinates, and the Perception RL reward is 1 if the coordinate in the response's tool call lies inside that box and 0 otherwise, including unparsable responses. Both RL teachers standardize rewards by the group mean and standard deviation, use a clipping ratio of 0.2, and add a low-variance KL loss toward the initial model. The Agentic RL teacher uses no KL term, yet its τ_T is 0.23 times that of its SFT counterpart, between the corresponding Reasoning and Perception ratios (0.19 and 0.37; Section 5.1).

B.3 Student Training Configurations

Table 5 reports the shared OPD settings for both student groups in each domain. Input batch counts prompts, and n is the number of sampled responses per prompt. The resulting optimizer batches contain 1,024, 32, and 112 sequences for Agentic, Reasoning, and Perception, respectively.

Table 5: Student OPD training settings.

Parameter	Agentic	Reasoning	Perception
Optimizer	AdamW	AdamW	AdamW
Learning rate	10^{-6}	1.875×10^{-6}	1.875×10^{-6}
Weight decay	0	0.01	0.01
Input batch	128	32	112
Responses per prompt (n)	8	1	1
Warmup (updates)	0	3	3
Max response tokens	512	8,192	128
Temperature	1.0	1.0	0.9
Top-p	1.0	0.95	0.9
Sampling top-k	Disabled	20	50

All student runs use optimizer betas $(0.9, 0.999)$, a constant learning rate after warmup, a gradient clipping norm of 1, and a maximum prompt length of 8,192 tokens. OPD uses no LoRA. Reasoning freezes the visual encoder and merger; Agentic and Perception freeze no parameters.

The sampling settings in Table 5 apply to training rollouts. All teachers remain fixed during OPD. Agentic follows the batch size, responses per prompt, sequence limits, learning rate, and rollout sampling settings of UI-MOPD's student training (Lian et al., 2026) and uses sampled-token distillation; Reasoning and Perception use full-vocabulary reverse KL. Each domain uses the same objective for both student groups.

B.4 Checkpoint Selection and Evaluation

Teacher checkpoints are selected to have comparable MobileWorld mean success rates for Agentic, Geo3K accuracies for Reasoning, and unweighted mean accuracies across six benchmarks for Perception. Student checkpoints are evaluated with the same procedures, and Appendix C reports all their benchmark results.

MobileWorld uses greedy decoding, at most 50 interaction steps, and up to four previous screenshots plus the current screenshot and action-history text.

Reasoning evaluates 601 Geo3K test problems, 1,000 MathVista testmini problems, 304 MathVision testmini problems, and 788 MathVerse testmini Vision-Only problems. Teachers and students generate one answer per problem with thinking disabled, temperature 1, top-p 0.95, top-k 20, presence penalty 1.5, and at most 8,192 output tokens in a 16,384-token context. We use the benchmark answer-scoring procedures; MathVista and MathVerse use Qwen3.5-122B-A10B for answer extraction or equivalence checking.

Perception evaluates ScreenSpot-v2 (1,272 examples), ScreenSpot-Pro (1,581), UI-Vision element grounding (5,479), MMBench-GUI L2 (3,594), OSWorld-G (564), and OSWorld-G-Refine (564).

ScreenSpot-Pro uses the default configuration of its official evaluation repository. For ScreenSpot-v2, UI-Vision, MMBench-GUI L2, and both OSWorld-G variants, decoding uses temperature 0.7, top-p 0.8, top-k 20, and at most 4,096 output tokens. MMBench-GUI L2 averages its Basic and Advanced accuracies equally. Cross-benchmark means weight each benchmark equally.

C Results Across Checkpoints

Tables 6–8 report the benchmark results of the student checkpoints shown in Figures 1–3, together with those of the base model and both teachers.

Table 6: MobileWorld results across OPD checkpoints. R1–R3: successes per round; Mean: success rate (%).

	SFT				RL			
Checkpoint	R1	R2	R3	Mean	R1	R2	R3	Mean
Teacher	28	23	29	22.79	26	28	28	23.36
Qwen3.5-9B	22	20	20	17.66	22	20	20	17.66
Step 8	21	28	21	19.94	17	23	25	18.52
Step 16	19	19	20	16.52	25	18	23	18.80
Step 24	23	22	22	19.09	23	29	24	21.65
Step 32	23	22	24	19.66	33	26	26	24.22
Step 40	19	24	21	18.23	30	29	25	23.93
Step 48	21	24	24	19.66	25	28	30	23.65
Step 56	22	22	21	18.52	31	29	24	23.93
Step 64	18	18	22	16.52	27	27	25	22.51

Table 7: Reasoning accuracy (%) across OPD checkpoints.

Model	Step	Geo3K	MathVista	MathVision	MathVerse	Avg.
Qwen3.5-9B	–	79.70	84.00	57.24	70.18	72.78
SFT teacher	–	88.35	80.80	53.95	67.89	72.75
RL teacher	–	84.19	83.60	57.57	73.10	74.61
Student ← SFT	4	86.36	83.60	51.32	71.45	73.18
	8	86.19	81.50	55.59	64.85	72.03
	12	89.02	82.10	56.91	68.15	74.04
	16	90.02	81.00	58.55	69.16	74.68
	20	83.86	82.10	54.61	69.16	72.43
	24	78.70	80.90	56.25	69.92	71.44
	28	81.20	82.40	58.88	71.32	73.45
	32	76.71	82.50	54.61	70.69	71.12
Student ← RL	4	82.03	82.10	56.58	70.05	72.69
	8	90.52	83.70	57.89	72.46	76.14
	12	89.85	84.30	58.22	74.37	76.68
	16	90.18	82.70	57.24	71.57	75.42
	20	91.51	83.10	58.55	71.07	76.06
	24	90.68	83.60	60.53	70.30	76.28
	28	89.68	83.90	57.24	70.81	75.41
	32	90.35	84.10	57.24	71.83	75.88

Table 8: Perception accuracy (%) across OPD checkpoints.

Model	Step	ScreenSpot-v2	ScreenSpot-Pro	UI-Vision	MMBench-GUI L2	OSWorld-G	OSWorld-G-R	Avg.
Qwen3.5-9B	–	91.75	63.00	26.92	80.41	61.35	67.73	65.19
SFT teacher	–	92.92	65.15	33.67	83.87	62.94	70.92	68.24
RL teacher	–	93.16	65.28	30.52	82.39	63.65	71.28	67.71
Student ← SFT	2	91.12	62.87	26.14	79.53	61.52	68.44	64.94
	4	91.35	63.19	27.52	81.17	61.52	68.44	65.53
	6	92.61	63.38	32.94	82.66	63.30	67.91	67.13
	8	93.24	63.76	38.38	84.32	65.60	71.10	69.40
	10	94.50	64.20	39.08	84.54	64.72	71.28	69.72
	20	94.18	64.83	40.28	85.28	65.78	71.81	70.36
	30	94.26	64.64	38.88	84.73	64.18	73.23	69.99
	40	93.63	64.96	39.04	85.12	65.60	71.63	70.00
	50	93.95	65.02	38.95	84.76	65.78	71.63	70.02
	60	94.34	64.20	37.32	84.41	64.01	72.16	69.41
	62	93.47	64.07	38.55	84.19	61.52	71.45	68.88
Student ← RL	2	91.43	62.49	26.26	79.91	58.51	65.60	64.03
	4	92.69	63.44	28.47	81.09	61.35	67.91	65.83
	6	92.77	65.34	33.51	83.81	63.12	70.74	68.21
	8	93.63	65.97	40.70	86.03	67.02	72.16	70.92
	10	94.89	65.97	42.31	86.54	65.60	71.99	71.22
	20	93.71	66.67	41.74	86.70	66.84	70.21	70.98
	30	94.03	66.10	38.09	85.26	63.48	70.74	69.62
	40	92.85	66.98	35.39	85.20	63.30	70.39	69.02
	50	93.71	66.54	35.97	84.70	63.30	71.45	69.28
	60	93.40	67.81	37.32	85.12	64.36	69.33	69.56
	62	93.63	66.60	35.48	84.65	64.36	69.50	69.04

D Further Directions

A natural extension is multi-teacher OPD, in which several domain teachers jointly supervise one student and domain mixing and teacher routing also come into play (Ma et al., 2026; Men et al., 2026; Lian et al., 2026). The smaller parameter displacements of RL teachers may make several such teachers easier to accommodate within one student. Our MobileWorld results further suggest that the choice of teacher-training algorithm may be particularly important for OPD in Agentic tasks. A valuable next step is to test whether this effect extends to more challenging agentic benchmarks, including long-horizon settings such as Long-Horizon-Terminal-Bench (Li et al., 2026c). Data synthesis methods that improve post-training data diversity (Li et al., 2026b) could further support studying how this effect varies across training distributions.