PROOPD: PROCESS-GUIDED ON-POLICY DISTILLATION FOR ITERATIVE TEXT-TO-IMAGE GENERATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Unified multimodal models (UMMs) can improve text-to-image generation through an iterative process that interleaves textual reasoning, image synthesis, reflection, and refinement, but training such policies faces two challenges: heterogeneous task objectives and the limited guidance that final-image rewards provide for intermediate decisions. We introduce ProOPD, a process-guided on-policy distillation framework for iterative text-to-image generation. A shared student produces its own text and image trajectories, and frozen stage-specific teachers provide dense supervision at the resulting token prefixes and visual sampling states. The supervision covers initial reasoning and generation as well as subsequent reflection and refinement, following the states encountered by the student as its policy changes. We combine textual distribution matching and visual velocity matching with image-level outcome optimization to train the generation process toward improved image quality. Experiments on four challenging text-to-image benchmarks (OneIG-EN, LongText-EN, WISE, and GenEval++) assess generation performance and the effectiveness of iterative refinement, with ablations examining the contributions of process supervision and outcome optimization.

1 INTRODUCTION

Complex text-to-image generation tasks are often difficult to fully satisfy with a single attempt (Ghosh et al., 2023; Li et al., 2024). Unified multimodal models (UMMs), such as BAGEL (Deng et al., 2025), provide a natural foundation for iterative generation by jointly supporting textual reasoning and visual synthesis within a single model. This enables a model to progressively improve its output through a sequence of prompt interpretation, image generation, discrepancy analysis, and visual refinement (Huang et al., 2025; 2026), as illustrated in Figure 1(a). Effectively exploiting this iterative process, however, requires these heterogeneous capabilities to be coordinated within a unified refinement policy.

A natural starting point is supervised fine-tuning (SFT) on demonstrations of the constituent tasks, including reasoning, generation, reflection, and refinement. However, we find that jointly fine-tuning a single model across all stages is insufficient for effectively integrating these heterogeneous capabilities. Specifically, we compare two SFT strategies under the same two-round generation-and-refinement workflow. In the specialist pipeline, each stage is handled by a separate model fine-tuned only on demonstrations for that stage. In the unified setting, a single model is jointly fine-tuned on demonstrations from all stages. The specialist pipeline achieves stronger results across multiple benchmarks (Figure 1(b) and Figure 3), revealing a capability-integration gap: joint SFT does not fully recover the performance achieved through stage-specific optimization.

Recent work has explored reinforcement learning (RL) to better integrate reasoning and generation capabilities through reward-guided self-exploration (Jiang et al., 2025). By sampling its own textual and visual actions and learning from the resulting images, a unified model can learn how to use these capabilities together during generation and refinement. This approach has been explored for both joint optimization of reasoning and visual synthesis (Liu et al., 2026a) and self-reflective generation (Ye et al., 2026; Huang et al., 2026). However, such training primarily relies on image-level outcome rewards, which evaluate the combined effects of multiple stages but provide limited direct guidance

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.

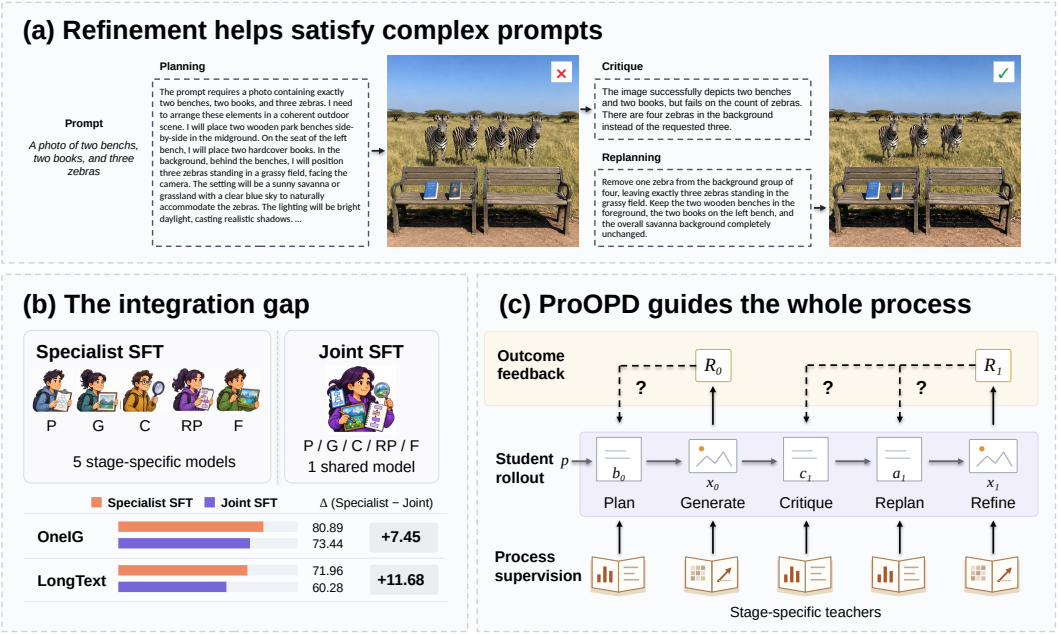

Figure 1: **Motivation and central idea of ProOPD.** (a) A generation-and-refinement example shows how critique and replanning correct an object-count error, reducing four zebras to the requested three while preserving the two benches and two books. Planning text is truncated for space. (b) A pipeline of separately fine-tuned specialists achieves higher scores than joint SFT, revealing a capability-integration gap (Figure 3). (c) ProOPD transfers independently optimized specialist capabilities into a shared student through stage-specific token distributions and visual velocity targets on student-generated trajectories. This process supervision complements image-level outcome feedback, whose guidance to intermediate textual decisions is indirect (dashed arrows).

for intermediate decisions. For example, a low-scoring image does not directly indicate whether the model should identify a different discrepancy, formulate a more actionable revision plan, or better execute the intended edit.

These challenges motivate stage-specific process supervision for learning a unified refinement policy. MOPD offers a relevant precedent by integrating specialized language-model capabilities into a shared student through on-policy distillation (OPD) from multiple teachers (Ma et al., 2026). For iterative generation, specialist teachers can provide targets suited to each operation, complementing the indirect guidance from image-level rewards. Crucially, OPD obtains these targets on the student's own evolving trajectories, where reasoning guides image synthesis and generated images and reflections condition subsequent revisions.

To train a unified policy with stage-specific guidance, we introduce ProOPD, a process-guided on-policy distillation framework for iterative text-to-image generation. Specifically, a shared student produces its own reasoning, images, reflections, and revision plans, while frozen stage-specific teachers supervise the corresponding operations (Figure 1(c)). For textual actions, teachers provide next-token distributions at student-generated prefixes; for image generation and refinement, they provide velocity targets at the student's visual sampling states. Textual distribution matching and visual velocity matching therefore supply dense supervision across both modalities. As new student rollouts are collected, the supervision follows the states visited by the evolving policy, integrating specialist guidance into a single refinement policy. To complement dense process supervision with image-level outcome feedback, ProOPD combines stage-specific distillation with reward-based optimization of the refinement policy. Teacher targets directly supervise intermediate textual and visual actions, while image-level rewards optimize the overall generation outcome. Both learning signals update the same student across all stages, jointly shaping its iterative refinement behavior. We evaluate process supervision and outcome optimization separately and jointly to examine their respective contributions.

We evaluate ProOPD on four challenging text-to-image benchmarks (OneIG-EN, LongText-EN, WISE, and GenEval++), assessing generation performance and the additional gains from iterative refinement. Compared with joint SFT, ProOPD improves second-round scores by 8.08 points on OneIG-EN text rendering, 5.99 points on LongText-EN, 5.00 points on WISE, and 12.50 points on GenEval++. Its OneIG-EN text-rendering score exceeds that of the specialist pipeline (81.52 versus 80.89), and its LongText-EN gap to the pipeline decreases from 11.68 to 5.69 points. Across these four metrics, ProOPD also achieves larger gains from the first to the second generation round than joint SFT.

Our contributions are as follows:

- We identify a capability-integration gap in iterative text-to-image generation: under the same generation-and-refinement workflow, joint SFT does not fully recover the performance of a pipeline of stage-specific specialists.
- We introduce ProOPD, a process-guided on-policy distillation framework that integrates capabilities optimized for heterogeneous stage objectives into a unified refinement policy. Stage-specific teachers are optimized independently, and a shared student learns from them on its own generation-and-refinement trajectories. Textual distribution matching and visual velocity matching both transfer specialist capabilities and directly supervise intermediate actions, while a joint process-and-outcome objective incorporates image-level reward-based policy optimization.
- Experiments show that ProOPD exceeds the specialist pipeline on OneIG-EN text rendering, narrows its gap to the pipeline on LongText-EN, and improves WISE and GenEval++ performance over joint SFT. Analysis of the first refinement round shows larger gains on these metrics. When all models refine the same initial images, students trained with OPD obtain higher OneIG-EN text-rendering, LongText-EN, and WISE scores than joint SFT. Training-objective comparisons examine the roles of process supervision and outcome optimization.

2 Related Work

Unified multimodal models. Unified multimodal models place image understanding and image generation in one model. They differ in how images are produced: one line predicts discrete visual tokens with the autoregressive objective also used for text (Chameleon Team, 2024; Wang et al., 2024; Chen et al., 2025), and another attaches a diffusion or flow-matching branch to the language backbone (Zhou et al., 2024; Shi et al., 2024; Xie et al., 2025). More recent work removes the external vision encoder and VAE and models pixels end-to-end (Diao et al., 2026b; Liu et al., 2026b). This unification gives the model access to its own output: it can read a generated image, state which prompt requirements are unmet, and condition a revision on that description. Reasoning, self-reflection, and refinement can therefore be carried out by one model along a single trajectory, without a separate evaluator or editing model. We use BAGEL, which couples an autoregressive backbone with a flow-matching image branch and supports understanding, generation, and image editing in one model (Deng et al., 2025).

Post-training for multimodal generation. Both supervised fine-tuning and reinforcement learning have been used to post-train models for image generation, with some methods also training a reflection-refinement pipeline. IRG follows an initial think-and-generate stage with reflection and visual refinement, and learns this behavior from decomposed supervision on complete trajectories (Huang et al., 2025). Reinforcement learning has been applied in several settings. Flow-GRPO converts deterministic flow-matching sampling into stochastic transitions so that group-based policy optimization applies to image generation (Liu et al., 2025a). T2I-R1 jointly optimizes semantic-level reasoning and token-level image generation (Jiang et al., 2025), and UniGRPO applies unified policy optimization to reasoning-driven visual generation (Liu et al., 2026a). R3 and AlphaGRPO apply reinforcement learning to reflection and refinement. R3 assigns separate rewards to its reflection and refinement stages (Ye et al., 2026), and AlphaGRPO uses decompositional verifiable rewards for semantic alignment and visual quality, reporting its results without a cold-start distillation stage (Huang et al., 2026). Across these methods, intermediate textual and visual actions are supervised either by fixed demonstrations or by rewards computed on the generated image.

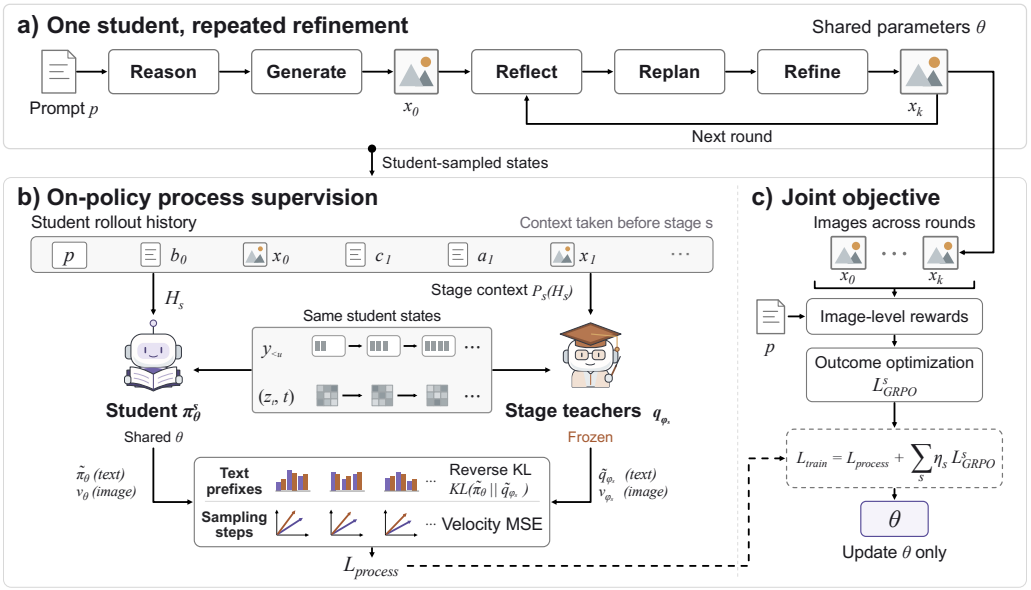

Figure 2: **Overview of ProOPD.** (a) One shared student performs generation and iterative refinement. (b) Frozen stage-specific teachers supervise the student's token prefixes and visual sampling states through reverse KL and velocity matching. (c) Process supervision and image-level GRPO jointly update only the student. The loop depicts the general iterative process; the experiments in this paper use initial generation followed by at most one refinement.

On-policy distillation. On-policy distillation trains a student on its own samples with token-level teacher feedback, reducing the mismatch between the prefixes seen during training and the states visited at inference (Agarwal et al., 2024). MOPD integrates domain-specialized language-model teachers into one student by supervising it on its own rollouts (Ma et al., 2026), and Flow-OPD supplies teacher velocity targets along student-sampled trajectories in flow-matching models (Fang et al., 2026). Each of these methods supervises one action type, either text tokens or visual sampling states. ProOPD assigns teachers to the stages of a single multimodal trajectory: textual teachers supervise planning, critique, and replanning at student-generated prefixes, and visual teachers supervise generation and refinement at the student's sampling states. The shared student combines this stage-specific supervision with image-level outcome optimization. The visual targets are velocity fields, which requires an image branch trained with flow matching; in a unified model that generates images as discrete visual tokens, the visual supervision would take the same form as the textual supervision.

3 Process-Guided On-Policy Distillation

Figure 2 summarizes the student rollout, stage-specific process supervision, and joint training objective.

3.1 Textual and Visual Actions in Generation

Given a text prompt p, the student first produces textual reasoning and a generation plan b_0, then synthesizes an initial image x_0. It subsequently generates a reflection c_1, an edit plan a_1, and a refined image x_1. The generation process is

$$\tau = (p, b_0, x_0, c_1, a_1, x_1). \tag{1}$$

Table 1: Teacher supervision within the generation process. All plans, reflections, and conditioning images in the teacher inputs come from the student rollout.

Stage	Teacher context	Supervision
Round 1 planning	Prompt p	Plan-token distribution
Round 1 generation	Prompt p, plan b_0	Synthesis velocity
Critique	Prompt p, plan b_0, image x_0	Reflection-token distribution
Replanning	Prompt p, plan b_0, image x_0, reflection c_1	Edit-plan-token distribution
Refinement	Prompt p, image x_0, edit plan a_1	Refinement velocity

One parameter set θ defines all five stage-conditioned policies:

$$\begin{aligned} b_0 &\sim \pi_\theta^{\text{plan}}(\cdot \mid p), & x_0 &\sim \pi_\theta^{\text{gen}}(\cdot \mid p, b_0), \\ c_1 &\sim \pi_\theta^{\text{crit}}(\cdot \mid p, b_0, x_0), \\ a_1 &\sim \pi_\theta^{\text{replan}}(\cdot \mid p, b_0, x_0, c_1), \\ x_1 &\sim \pi_\theta^{\text{ref}}(\cdot \mid p, b_0, x_0, c_1, a_1). \end{aligned} \tag{2}$$

The superscripts identify operations performed by the shared student. Reflection can terminate the process with an explicit decision that refinement is unnecessary; in that case, replanning and refinement are skipped and the final output is x_0. Otherwise, the final output is x_1. Both initial generation and subsequent refinement participate in training.

3.2 Stage-specific teachers and student rollouts

We train and freeze teachers for initial planning, image generation, critique, replanning, and refinement. Each teacher learns from supervision appropriate to its operation. For stage s, let H_s denote the student's conditioning context and $P_s(H_s)$ the context presented to the teacher. The projection uses the relevant student outputs in the teacher's task format. Table 1 summarizes this correspondence. The teachers guide training, while inference uses the student alone.

During training, we sample G initial plan–image pairs $\{(b_0^i, x_0^i)\}_{i=1}^G$ for each prompt. We select the pair with the lowest image reward as the common source for G independently sampled reflection and refinement branches:

$$j^* = \operatorname*{arg\,min}_{i \in \{1,\dots,G\}} R(p, x_0^i). \tag{3}$$

The selected plan and image are carried together into each branch. Initial-stage losses use the original candidates; subsequent-stage losses use branches conditioned on the selected source. This concentrates refinement training on less successful student outputs. The same source-selection procedure is used in the reward-only, distillation-only, and combined training comparisons. At benchmark inference, each trajectory refines its own initial image.

Let $\mathcal{B} \sim \mathcal{R}_{\bar{\theta}}$ denote a batch produced by this sampling and selection procedure, with $\bar{\theta}$ the student parameters used for rollout. Teacher targets are evaluated on the sampled states, and these states remain fixed during the update. New rollouts refresh the supervision distribution as the student changes.

3.3 Process supervision across modalities

For a textual stage, the teacher provides a next-token distribution at each student-generated prefix $y_{<u}$. We minimize reverse KL from the student to the teacher:

$$\ell_{s,u}^{\text{text}} = D_{\text{KL}}(\widetilde{\pi}_\theta^s(\cdot \mid H_s, y_{<u}) \,\|\, \widetilde{q}_{\phi_s}(\cdot \mid P_s(H_s), y_{<u})) \,. \tag{4}$$

The grouped distributions $\widetilde{\pi}$ and $\widetilde{q}$ retain the teacher's highest-probability tokens and aggregate the remaining probability mass into one category; their construction is given in Appendix A.2. This provides distributional guidance for initial reasoning, reflection, and replanning at the prefixes produced by the student.

For image generation and refinement, we match the student and teacher velocity fields at the same student-sampled latent z_t and time t:

$$\ell_{s,t}^{\text{img}} = \frac{1}{d} \left\| v_\theta^s(z_t, t \mid H_s) - v_{\phi_s}(z_t, t \mid P_s(H_s)) \right\|_2^2, \tag{5}$$

where d is the latent dimensionality and the velocities include the respective classifier-free guidance settings. Teacher targets are detached. Supervision uses the latent states saved from the student's sampling trajectory, covering all image-sampling transitions.

Let $D_s(\theta; \mathcal{B})$ be the reduced loss for stage s, obtained from the corresponding token or image-transition losses. The process objective is

$$\mathcal{L}_{\text{process}}(\theta; \bar{\theta}) = \mathbb{E}_{\mathcal{B} \sim \mathcal{R}_{\bar{\theta}}} \left[\sum_{s \in \mathcal{S}} \lambda_s D_s(\theta; \mathcal{B}) \right]. \tag{6}$$

The weights λ_s balance supervision across stages and modalities. Text losses are averaged over valid tokens; image losses are averaged over latent dimensions and sampling transitions. Appendix A.2 specifies the batch reduction and treatment of stages skipped after stopping.

3.4 Joint process and outcome optimization

Image-level feedback evaluates semantic alignment, text rendering, and visual quality. We use decomposed prompt checks and OCR to construct image rewards, and apply GRPO-style policy optimization (Shao et al., 2024; Huang et al., 2026). Initial planning and generation use feedback from the initial images; reflection, replanning, and refinement use feedback from the corresponding final outputs. Advantages are computed within each prompt group, with an additional repetition penalty for textual actions.

The complete objective combines process guidance with outcome optimization:

$$\mathcal{L}_{\text{train}} = \mathcal{L}_{\text{process}} + \sum_{s \in \mathcal{S}} \eta_s \mathcal{L}_{\text{GRPO}}^s, \tag{7}$$

where η_s weights the policy loss for stage s. Textual policy losses act on sampled tokens, and visual policy losses act on the stochastic image-sampling transitions. The image distillation loss also supervises the deterministic transitions.

Process supervision supplies local targets for textual and visual actions, while the outcome objective trains the generation process toward prompt satisfaction and image quality. We compare the two objectives separately and jointly.

4 Experiments

4.1 Experimental setting

Data and initialization. We use BAGEL as the unified multimodal backbone (Deng et al., 2025). Training prompts cover compositional constraints, knowledge and reasoning, text rendering, and visual appearance, with verifiable checks for image evaluation. Reference images and their associated plans provide supervision for initial reasoning and generation. For reflection and refinement, we annotate BAGEL-generated initial images with critiques and edit plans, and retain accepted refined images as visual targets. We train five stage-specific teacher adapters and initialize a single student adapter by supervised fine-tuning across the five stages. The teachers remain frozen during on-policy training. Data construction and training settings are given in Appendix A.

Benchmarks and inference. We evaluate OneIG-EN (Chang et al., 2025), LongText-EN (X-Omni, 2025), WISE_Verified (Niu et al., 2025) (referred to as WISE below), and GenEval++ (Ye et al., 2025). OneIG reports alignment, text rendering, reasoning, style, and diversity; LongText evaluates rendered text; WISE evaluates knowledge-dependent semantic generation; GenEval++ tests instruction following under more complex compositional constraints. Scores are displayed on a 0–100 scale. All student and specialist comparisons use four independent rollouts per prompt

Table 2: Comparison with image generation models and UMMs on four benchmarks. Joint SFT and ProOPD report outputs after refinement. †Previously reported results.

Model	OneIG-EN					LongText-EN↑	WISE↑	GenEval++↑
	Align.↑	Text↑	Reason.↑	Style↑	Div.↑			
Image generation models								
SD3.5-Large	80.9†	62.9†	29.4†	35.3†	22.5†	–	40.4†	–
FLUX.1-dev	78.6†	52.3†	25.3†	36.8†	23.8†	60.7†	41.6†	31.4†
Z-Image	88.1†	98.7†	28.0†	38.7†	19.4†	93.5†	45.3†	–
GPT-Image-2	90.3†	74.1†	35.2†	44.3†	13.9†	96.4†	85.4	–
Unified multimodal models								
Janus-Pro-7B	55.3†	0.1†	13.9†	27.6†	36.5†	1.9†	33.4†	24.6†
Show-o2-7B	81.7†	0.2†	22.6†	31.7†	17.7†	0.6†	–	–
BAGEL	80.97	38.75	19.08	38.55	18.45	36.28	37.70	36.79
BAGEL w/ self-CoT	80.14	6.10	22.08	38.98	18.23	6.71	65.70	46.43
IRG	83.9†	37.7†	23.9†	42.7†	19.2†	–	–	–
R3	–	–	–	–	–	–	–	68.9†
AlphaGRPO (SRR)	82.59	2.59	21.23	42.38	20.81	4.16	63.00	57.50
Joint SFT	83.78	73.44	32.52	44.09	20.84	60.28	64.70	62.50
ProOPD	85.18	81.52	32.38	43.76	18.96	66.27	69.70	75.00

on OneIG and LongText, and one on WISE and GenEval++. The WISE scores for both the 10k-update SFT initialization and the specialist pipeline come from earlier single-rollout evaluations with greedy text decoding, whereas the post-trained students sample text at temperature 0.3. We report Round 1 outputs from reasoning and generation and Round 2 outputs after reflection and optional refinement. When refinement is skipped, Round 2 retains the Round 1 image. The main comparisons use 1024×1024 images and allow at most one refinement. Sampling and scoring details are provided in Appendix A.4.

4.2 Comparison with image generation models and UMMs

Table 2 organizes the model comparison into image generation models and unified multimodal models. For ProOPD, we report Round 2 outputs after reflection and optional refinement, requiring up to two image-generation calls per prompt. ProOPD uses the GRPO + OPD checkpoint after 160 post-training updates.

Generation performance. ProOPD reaches 81.52 on OneIG text rendering, 66.27 on LongText, 69.70 on WISE, and 75.00 on GenEval++, exceeding the evaluated BAGEL, BAGEL with self-CoT, and AlphaGRPO (SRR) baselines on all four metrics. Relative to the Joint SFT initialization, its OneIG text-rendering, LongText, and GenEval++ scores increase by 8.08, 5.99, and 12.50 points, respectively. The shared student thus improves beyond supervised capability integration, with gains in both text rendering and compositional instruction following. Table 3 isolates the contributions of the two post-training objectives under a shared initialization and rollout procedure.

4.3 Process supervision and outcome optimization

Table 3 ablates process supervision and outcome optimization after 160 post-training updates, using Round 2 outputs. Removing process supervision gives reward-only GRPO; removing outcome optimization gives Pure OPD. All runs start from the same 10k-update supervised student and use the same training prompts and group-based source selection. Pure OPD uses image rewards to select the refinement source but has no policy-gradient loss.

Process supervision beyond image-level rewards. Adding process supervision to GRPO improves LongText by 3.66 points (66.27 versus 62.61), WISE by 4.00 points (69.70 versus 65.70),

Table 3: Ablation of process supervision and outcome optimization. GRPO removes process supervision; Pure OPD removes the outcome policy-gradient loss. Bold marks the highest score in each column.

Objective	OneIG-EN					LongText	WISE	GenEval++
	Align.	Text	Reason.	Style	Div.			
GRPO	84.27	80.80	**32.65**	43.17	**21.43**	62.61	65.70	75.00
Pure OPD	84.80	79.57	32.22	43.52	18.60	65.96	68.00	**75.36**
ProOPD	**85.18**	**81.52**	32.38	**43.76**	18.96	**66.27**	**69.70**	75.00

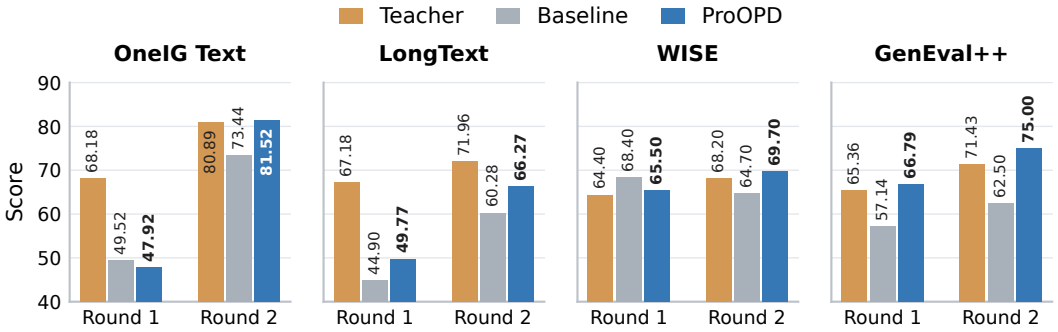

Figure 3: Generation and refinement across two rounds. Teacher uses stage-specific specialists; Baseline is Joint SFT.

and OneIG alignment by 0.91 points (85.18 versus 84.27). These results support the value of stage-specific teacher guidance beyond image-level outcome optimization. Matched update counts do not imply matched training compute, because OPD additionally evaluates the teachers.

Combining process supervision and outcome optimization. Adding outcome optimization to Pure OPD improves OneIG text rendering by 1.95 points (81.52 versus 79.57) and WISE by 1.70 points (69.70 versus 68.00). At this checkpoint, the combined objective achieves the highest scores among the three objectives on OneIG alignment, text rendering, and style, as well as LongText and WISE. GenEval++ scores remain close, ranging from 75.00 to 75.36. Together, these comparisons support the complementary roles of intermediate teacher guidance and image-level outcome feedback.

4.4 Capability Integration and Iterative Refinement

Figure 3 compares the specialist teachers, the SFT baseline, and ProOPD across two rounds on text rendering, knowledge-dependent generation, and compositional instruction following. The Teacher pipeline uses the same frozen stage-specific adapters as the distillation teachers, while Baseline and ProOPD each use one shared adapter throughout the process.

Integrating specialist capabilities. In Round 2, Teacher exceeds Baseline on LongText (71.96 versus 60.28), OneIG text rendering (80.89 versus 73.44), WISE (68.20 versus 64.70), and GenEval++ (71.43 versus 62.50). ProOPD reduces the LongText gap from 11.68 to 5.69 points and exceeds Teacher on OneIG text rendering (81.52 versus 80.89), WISE (69.70), and GenEval++ (75.00). These results show that the shared student closes the gap to the specialist pipeline on these tasks, although a gap remains on LongText.

Additional comparisons with shared initial images are provided in Appendix A.5.

Alignment and constraint preservation. For ProOPD, refinement changes OneIG alignment from 85.98 to 85.18, a decrease of 0.80 points, compared with a decrease of 2.05 points for Baseline. Post-refinement style and diversity scores also remain below Baseline (43.76 versus 44.09, and 18.96 versus 20.84).

4.5 Performance across refinement rounds

We also compare initial-generation scores with the Round 2 results in Figure 3.

From Round 1 to Round 2, ProOPD gains 33.60 points on OneIG text rendering, 16.50 points on LongText, 4.20 points on WISE, and 8.21 points on GenEval++, compared with changes of 23.92, 15.38, -3.70, and 5.36 points for Baseline. For Baseline, refinement lowers WISE from 68.40 to 64.70. These results establish the observed gains from the first refinement, but do not determine whether further iterations continue to improve performance, reach a plateau, or introduce regressions.

4.6 Instruction-based image editing

We evaluate instruction-based editing on GEdit-Bench-EN (Liu et al., 2025b): 606 instructions across 11 task types. BAGEL's native editing context produces one edited image without planning or reflection. GPT-4.1 evaluates semantic consistency (SC) and perceptual quality (PQ) using VIEScore (Ku et al., 2024). Overall is $\sqrt{\mathrm{SC} \cdot \mathrm{PQ}}$ per sample, macro-averaged across task types. ProOPD scores 8.20/7.20/7.42 in SC/PQ/O. Published AlphaGRPO (SRR) reaches 7.08 in O in a separate evaluation. Protocol details are in Appendix A.4.

Table 4: Single-step image editing on GEdit-Bench-EN. †Previously reported results from Huang et al. (2026).

Model	SC↑	PQ↑	O↑
GPT-4o	7.85†	7.62†	7.53†
OmniGen2	7.16†	6.77†	6.41†
Step1X-Edit	7.09†	6.76†	6.70†
BAGEL	7.36†	6.83†	6.52†
AlphaGRPO (RT2I)	7.54†	7.18†	6.85†
AlphaGRPO (SRR)	7.67†	7.46†	7.08†
ProOPD	8.20	7.20	7.42

5 Limitations and Discussion

Process supervision depends on the quality of the stage-specific teachers. A teacher may provide an unsuitable target for a student-generated state, and agreement with an individual teacher does not ensure that the resulting image better satisfies the prompt. Outcome optimization complements these local targets, while its own reliability depends on the image evaluators and their treatment of semantic constraints, text rendering, and visual quality.

Teacher evaluations add training cost. The implementation shares a backbone across teacher adapters, but each supervised stage still requires additional teacher computation. At inference, textual reasoning and reflection consume tokens, and refinement requires an additional image-generation call when executed. Image quality and improvement should therefore be considered together with the inference budget.

6 Conclusion

We presented ProOPD, which trains a single UMM to plan, generate, critique, replan, and refine images. Frozen stage-specific teachers supervise the student's own text prefixes and visual sampling states, while image-level rewards guide the generated outcomes. After refinement, ProOPD improves OneIG text rendering, LongText, WISE, and GenEval++ over Joint SFT. The objective ablations support the complementary roles of process supervision and outcome optimization, and the two-round comparison shows gains from refinement. Together, these results show how specialist guidance can be integrated into one policy that improves its own generations through critique and refinement.

AI use statement

Generative AI tools assisted with literature search and language polishing. Model-based prompt construction, supervision synthesis, image generation, and evaluation are part of the experimental workflow described in the paper.

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. In *International Conference on Learning Representations*, 2024. doi: 10.48550/arXiv.2306.13649. URL https://arxiv.org/abs/2306.13649.

Chameleon Team. Chameleon: Mixed-Modal Early-Fusion Foundation Models. *arXiv preprint arXiv:2405.09818*, 2024. doi: 10.48550/arXiv.2405.09818. URL https://arxiv.org/abs/2405.09818.

Jingjing Chang, Yixiao Fang, Peng Xing, Shuhan Wu, Wei Cheng, Rui Wang, Xianfang Zeng, Gang Yu, and Hai-Bao Chen. OneIG-Bench: Omni-dimensional Nuanced Evaluation for Image Generation. *arXiv preprint arXiv:2506.07977*, 2025. doi: 10.48550/arXiv.2506.07977. URL https://arxiv.org/abs/2506.07977.

Xiaokang Chen, Zhiyu Wu, Xingchao Liu, Zizheng Pan, Wen Liu, Zhenda Xie, Xingkai Yu, and Chong Ruan. Janus-Pro: Unified Multimodal Understanding and Generation with Data and Model Scaling. *arXiv preprint arXiv:2501.17811*, 2025. doi: 10.48550/arXiv.2501.17811. URL https://arxiv.org/abs/2501.17811.

Chaorui Deng, Deyao Zhu, Kunchang Li, Chenhui Gou, Feng Li, Zeyu Wang, Shu Zhong, Weihao Yu, Xiaonan Nie, Ziang Song, Guang Shi, and Haoqi Fan. Emerging Properties in Unified Multimodal Pretraining. *arXiv preprint arXiv:2505.14683*, 2025. doi: 10.48550/arXiv.2505.14683. URL https://arxiv.org/abs/2505.14683.

Haiwen Diao, Jiahao Wang, Chenjing Ding, et al. SenseNova-U1.5: Towards Native Unified Visual Intelligence. *arXiv preprint arXiv:2609.11929*, 2026a. doi: 10.48550/arXiv.2609.11929. URL https://arxiv.org/abs/2609.11929.

Haiwen Diao, Penghao Wu, Hanming Deng, Jiahao Wang, Shihao Bai, Silei Wu, Weichen Fan, Wenjie Ye, Wenwen Tong, Xiangyu Fan, et al. SenseNova-U1: Unifying Multimodal Understanding and Generation with NEO-unify Architecture. *arXiv preprint arXiv:2605.12500*, 2026b. doi: 10.48550/arXiv.2605.12500. URL https://arxiv.org/abs/2605.12500.

Zhen Fang, Wenxuan Huang, Yu Zeng, Yiming Zhao, Shuang Chen, Kaituo Feng, Yunlong Lin, Lin Chen, Zehui Chen, Shaosheng Cao, and Feng Zhao. Flow-OPD: On-Policy Distillation for Flow Matching Models. *arXiv preprint arXiv:2605.08063*, 2026. doi: 10.48550/arXiv.2605.08063. URL https://arxiv.org/abs/2605.08063.

Dhruba Ghosh, Hanna Hajishirzi, and Ludwig Schmidt. GenEval: An Object-Focused Framework for Evaluating Text-to-Image Alignment. *arXiv preprint arXiv:2310.11513*, 2023. doi: 10.48550/arXiv.2310.11513. URL https://arxiv.org/abs/2310.11513.

Runhui Huang, Jie Wu, Rui Yang, Zhe Liu, and Hengshuang Zhao. AlphaGRPO: Unlocking Self-Reflective Multimodal Generation in UMMs via Decompositional Verifiable Reward. *arXiv preprint arXiv:2605.12495*, 2026. doi: 10.48550/arXiv.2605.12495. URL https://arxiv.org/abs/2605.12495.

Wenxuan Huang, Shuang Chen, Zheyong Xie, Shaosheng Cao, Shixiang Tang, Yufan Shen, Qingyu Yin, Wenbo Hu, Xiaoman Wang, Yuntian Tang, Junbo Qiao, Yue Guo, Yao Hu, Zhenfei Yin, Philip Torr, Yu Cheng, Wanli Ouyang, and Shaohui Lin. Interleaving Reasoning for Better Text-to-Image Generation. *arXiv preprint arXiv:2509.06945*, 2025. doi: 10.48550/arXiv.2509.06945. URL https://arxiv.org/abs/2509.06945.

Dongzhi Jiang, Ziyu Guo, Renrui Zhang, Zhuofan Zong, Hao Li, Le Zhuo, Shilin Yan, Pheng-Ann Heng, and Hongsheng Li. T2I-R1: Reinforcing Image Generation with Collaborative Semantic-level and Token-level CoT. *arXiv preprint arXiv:2505.00703*, 2025. doi: 10.48550/arXiv.2505.00703. URL https://arxiv.org/abs/2505.00703.

Max Ku, Dongfu Jiang, Cong Wei, Xiang Yue, and Wenhu Chen. VIEScore: Towards Explainable Metrics for Conditional Image Synthesis Evaluation. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 12268–12290.

Association for Computational Linguistics, 2024. doi: 10.18653/v1/2024.acl-long.663. URL https://aclanthology.org/2024.acl-long.663/.

Baiqi Li, Zhiqiu Lin, Deepak Pathak, Jiayao Li, Yixin Fei, Kewen Wu, Tiffany Ling, Xide Xia, Pengchuan Zhang, Graham Neubig, and Deva Ramanan. GenAI-Bench: Evaluating and Improving Compositional Text-to-Visual Generation. *arXiv preprint arXiv:2406.13743*, 2024. doi: 10.48550/arXiv.2406.13743. URL https://arxiv.org/abs/2406.13743.

Jie Liu, Gongye Liu, Jiajun Liang, Yangguang Li, Jiaheng Liu, Xintao Wang, Pengfei Wan, Di Zhang, and Wanli Ouyang. Flow-GRPO: Training Flow Matching Models via Online RL. *arXiv preprint arXiv:2505.05470*, 2025a. doi: 10.48550/arXiv.2505.05470. URL https://arxiv.org/abs/2505.05470.

Jie Liu, Zilyu Ye, Linxiao Yuan, Shenhan Zhu, Yu Gao, Jie Wu, Kunchang Li, Xionghui Wang, Xiaonan Nie, Weilin Huang, and Wanli Ouyang. UniGRPO: Unified Policy Optimization for Reasoning-Driven Visual Generation. *arXiv preprint arXiv:2603.23500*, 2026a. doi: 10.48550/arXiv.2603.23500. URL https://arxiv.org/abs/2603.23500.

Shiyu Liu, Yucheng Han, Peng Xing, Fukun Yin, Rui Wang, Wei Cheng, Jiaqi Liao, Yingming Wang, Honghao Fu, Chunrui Han, Guopeng Li, Yuang Peng, Quan Sun, Jingwei Wu, Yan Cai, Zheng Ge, Ranchen Ming, Lei Xia, Xianfang Zeng, Yibo Zhu, Binxing Jiao, Xiangyu Zhang, Gang Yu, and Daxin Jiang. Step1X-Edit: A Practical Framework for General Image Editing. *arXiv preprint arXiv:2504.17761*, 2025b. doi: 10.48550/arXiv.2504.17761. URL https://arxiv.org/abs/2504.17761.

Zhiheng Liu, Weiming Ren, Xiaoke Huang, Shoufa Chen, Tianhong Li, Mengzhao Chen, Yatai Ji, Sen He, Jonas Schult, Belinda Zeng, Tao Xiang, Wenhu Chen, Ping Luo, Luke Zettlemoyer, and Yuren Cong. Tuna-2: Pixel Embeddings Beat Vision Encoders for Multimodal Understanding and Generation. *arXiv preprint arXiv:2604.24763*, 2026b. doi: 10.48550/arXiv.2604.24763. URL https://arxiv.org/abs/2604.24763.

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. *arXiv preprint arXiv:2606.30406*, 2026. doi: 10.48550/arXiv.2606.30406. URL https://arxiv.org/abs/2606.30406.

Yuwei Niu, Munan Ning, Mengren Zheng, Weiyang Jin, Bin Lin, Peng Jin, Jiaqi Liao, Chaoran Feng, Fanqing Meng, Kunpeng Ning, Bin Zhu, and Li Yuan. WISE: A World Knowledge-Informed Semantic Evaluation for Text-to-Image Generation. *arXiv preprint arXiv:2503.07265*, 2025. doi: 10.48550/arXiv.2503.07265. URL https://arxiv.org/abs/2503.07265.

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. *arXiv preprint arXiv:2402.03300*, 2024. doi: 10.48550/arXiv.2402.03300. URL https://arxiv.org/abs/2402.03300.

Weijia Shi, Xiaochuang Han, Chunting Zhou, Weixin Liang, Xi Victoria Lin, Luke Zettlemoyer, and Lili Yu. LMFusion: Adapting Pretrained Language Models for Multimodal Generation. *arXiv preprint arXiv:2412.15188*, 2024. doi: 10.48550/arXiv.2412.15188. URL https://arxiv.org/abs/2412.15188.

Xinlong Wang, Xiaosong Zhang, Zhengxiong Luo, Quan Sun, Yufeng Cui, Jinsheng Wang, Fan Zhang, Yueze Wang, Zhen Li, Qiying Yu, Yingli Zhao, Yulong Ao, Xuebin Min, Tao Li, Boya Wu, Bo Zhao, Bowen Zhang, Liangdong Wang, Guang Liu, Zheqi He, Xi Yang, Jingjing Liu, Yonghua Lin, Tiejun Huang, and Zhongyuan Wang. Emu3: Next-Token Prediction is All You Need. *arXiv preprint arXiv:2409.18869*, 2024. doi: 10.48550/arXiv.2409.18869. URL https://arxiv.org/abs/2409.18869.

X-Omni. LongText-Bench. Hugging Face dataset, 2025. URL https://huggingface.co/datasets/X-Omni/LongText-Bench. Accessed September 8, 2026.

Jinheng Xie, Zhenheng Yang, and Mike Zheng Shou. Show-o2: Improved Native Unified Multimodal Models. *arXiv preprint arXiv:2506.15564*, 2025. doi: 10.48550/arXiv.2506.15564. URL https://arxiv.org/abs/2506.15564.

Junyan Ye, Dongzhi Jiang, Zihao Wang, Leqi Zhu, Zhenghao Hu, Zilong Huang, Jun He, Zhiyuan Yan, Jinghua Yu, Hongsheng Li, Conghui He, and Weijia Li. Echo-4o: Harnessing the Power of GPT-4o Synthetic Images for Improved Image Generation. *arXiv preprint arXiv:2508.09987*, 2025. doi: 10.48550/arXiv.2508.09987. URL https://arxiv.org/abs/2508.09987.

Sen Ye, Mengde Xu, Shuyang Gu, Di He, Liwei Wang, and Han Hu. Understanding vs. Generation: Navigating Optimization Dilemma in Multimodal Models. *arXiv preprint arXiv:2602.15772*, 2026. doi: 10.48550/arXiv.2602.15772. URL https://arxiv.org/abs/2602.15772.

Z-Image Team, Huanqia Cai, Sihan Cao, Ruoyi Du, Peng Gao, Aiming Hao, Steven Hoi, Zhaohui Hou, Shijie Huang, Dengyang Jiang, Yuming Jiang, Xin Jin, Liangchen Li, Zhen Li, Zhong-Yu Li, David Liu, Dongyang Liu, Qilong Wu, Feng Yu, Zechao Zhan, Chi Zhang, Shifeng Zhang, Ruikai Zhou, and Shilin Zhou. Z-Image: An Efficient Image Generation Foundation Model with Single-Stream Diffusion Transformer. *arXiv preprint arXiv:2511.22699*, 2025. doi: 10.48550/arXiv.2511.22699. URL https://arxiv.org/abs/2511.22699.

Chunting Zhou, Lili Yu, Arun Babu, Kushal Tirumala, Michihiro Yasunaga, Leonid Shamis, Jacob Kahn, Xuezhe Ma, Luke Zettlemoyer, and Omer Levy. Transfusion: Predict the Next Token and Diffuse Images with One Multi-Modal Model. *arXiv preprint arXiv:2408.11039*, 2024. doi: 10.48550/arXiv.2408.11039. URL https://arxiv.org/abs/2408.11039.

A Implementation and Evaluation Details

A.1 Data construction and supervised initialization

Prompt construction spans 15 capability axes, including object attributes, counting, spatial relations, composition, knowledge, reasoning, text rendering, and visual appearance. Prompts are generated with templates and language models, paired with verifiable conditions, and filtered using schema validation, quality checks, and duplicate detection against the training pool and benchmark prompts. The supervised prompt pool contains 116,679 prompts. Online training uses 42,342 prompts, with a separate held-out set of 512 prompts. Training prompts are selected from the training split using a reward range of at least 0.10 across eight initial image candidates.

Planning and generation supervision comes from reference images and their associated textual plans after prompt-satisfaction and image-quality filtering. Critique data includes both images needing correction and images requiring no refinement. Replanning targets specify explicit edits for problem cases, and refinement targets use edits accepted by pairwise image evaluation. The refinement teacher also receives reference-image generation examples.

All teachers and the student use the BAGEL backbone with LoRA rank 128 and scaling parameter 256. The five teachers use separate adapters; the student shares one adapter across stages. The supervised student uses stage-sampling weights $1:3:1:1:3$ for planning, generation, critique, replanning, and refinement, with loss $0.25\mathcal{L}_{\mathrm{CE}} + \mathcal{L}_{\mathrm{MSE}}$. On-policy training starts from its EMA weights after 10,000 supervised updates. The planning, generation, critique, replanning, and refinement teacher adapters are selected after 3,000, 8,000, 5,000, 5,000, and 8,000 updates, respectively.

The specialist pipeline in Figure 3 uses these same five stage-specific teacher checkpoints: planning at 3,000 updates, generation at 8,000, critique at 5,000, replanning at 5,000, and refinement at 8,000.

A.2 Distillation and loss reduction

For textual distillation, let q_v and π_v be the teacher and student probabilities after temperature scaling, normalized over the full vocabulary. Let V_k contain the teacher's top-k tokens, and define $\pi_{\mathrm{tail}} = 1 - \sum_{v \in V_k} \pi_v$ and $q_{\mathrm{tail}} = 1 - \sum_{v \in V_k} q_v$. The grouped reverse KL in Equation 4 is

$$\sum_{v \in V_k} \pi_v \log \frac{\pi_v}{q_v} + \pi_{\mathrm{tail}} \log \frac{\pi_{\mathrm{tail}}}{q_{\mathrm{tail}}}. \tag{8}$$

We use $k = 32$ and temperature 0.7, without an additional temperature-squared multiplier. The top-k probabilities retain their full-vocabulary mass.

Text losses are averaged over all valid tokens in an executed stage and multiplied by the fraction of trajectories executing that stage. Image losses are averaged over latent dimensions and sampling transitions, then summed over executed images and divided by the original batch size. A naturally sampled end-of-sequence token is supervised; an end marker inserted after length truncation is not a sampled action. Stopping retains the critique action and gives zero loss to skipped replanning and refinement stages.

The stage weights in Equation 7 are:

Stages	Process weight λ_s	GRPO weight η_s
Planning, critique, replanning	0.05	0.1
Generation, refinement	0.1	10

A.3 Rollouts and image rewards

On-policy training uses 64 NVIDIA H800 GPUs and takes approximately two days.

We use $G = 8$ candidates per prompt and an AdamW learning rate of 5×10^{-5}. Image rollouts use 1024×1024 resolution and 20 time points, giving 19 transitions. Five consecutive transitions selected within the first ten are stochastic, with noise level 0.7; the remaining transitions are deterministic. Image OPD averages velocity MSE over all 19 transitions. Image GRPO averages its

policy loss over the five stochastic transitions. Text sampling uses temperature 0.7, full-support sampling, and a maximum of 768 tokens per stage.

Raw image reward is the geometric mean of semantic and quality scores, following decompositional image evaluation (Huang et al., 2026). OCR-based scores replace text-specific semantic checks. For text-rendering prompts, OCR receives half the semantic weight; for other prompts with at least 100 non-whitespace target characters, its weight is at least one half. Otherwise, it retains the weight of the replaced text checks. When no other semantic checks remain, OCR supplies the full semantic score. The raw image reward selects the refinement source.

For GRPO, each check is standardized across the prompt group before the weighted semantic and quality checks are aggregated. The resulting image scores are normalized within the group to form advantages, separately for Round 1 and Round 2 outputs. For textual actions, a detected repetition subtracts 0.1 from the aggregated score before the final advantage normalization. The yes/no and OCR training judges use Qwen3.7-Flash; the textual repetition judge uses Qwen3.8-Flash. Pure OPD uses the Round 1 image reward only for source selection and does not apply GRPO or a repetition penalty to its loss.

A critique ending with the explicit sentence *No refinement is needed.* stops refinement. The Round 2 output and its image reward are inherited from the source image. Stopped samples remain in the prompt group for advantage computation.

A.4 Benchmark sampling and scoring

The GEdit-Bench-EN result in Table 4 uses the GRPO + OPD checkpoint after 160 post-training updates.

Table 5 lists the evaluation sample counts. Independent trajectories supply the multiple images for OneIG diversity; Round 1 and Round 2 images from a single trajectory are not treated as separate diversity samples.

In Table 2, entries marked † are reported by the benchmark papers, official leaderboards, or method papers (Chang et al., 2025; X-Omni, 2025; Niu et al., 2025; Ye et al., 2025; Huang et al., 2025; Ye et al., 2026; Z-Image Team et al., 2025; Diao et al., 2026a). GPT-Image-2's OneIG-EN and LongText-EN results are from Tables 8 and 12 of Diao et al. (2026a); its WISE result is evaluated by us. The WISE leaderboard uses the same judge as our evaluation. BAGEL and BAGEL with self-CoT each use one image-generation call, while AlphaGRPO (SRR) reports the output of its refinement stage. Joint SFT and ProOPD report Round 2 outputs, retaining the initial image when refinement is skipped.

Table 5: Sampling protocol for the student and specialist comparisons.

Benchmark	Prompts	Trajectories per prompt
WISE	1,000	1
LongText-EN	160	4
OneIG-EN	1,120	4
GenEval++	280	1

For the completed evaluations, inference uses 50 visual sampling time points, text temperature 0.3, and a 768-token limit per textual stage. Each trajectory evaluates its own Round 1 and Round 2 outputs without best-of-G selection. Teacher inputs use stage-specific task templates, while student inputs use shared progressive prefixes. The specialist and Joint SFT evaluations use paired image noise on OneIG and LongText; the post-trained student evaluations use separate noise draws. The WISE evaluations of the specialist pipeline and Joint SFT use greedy text decoding, whereas the post-trained students sample text at temperature 0.3 on all benchmarks.

WISE uses Qwen3.5-35B-A3B to produce binary judgments, averaged over the 1,000 examples. LongText-EN uses Qwen2.5-VL-7B-Instruct for OCR and aggregates matched ground-truth words over the total number of ground-truth words. OneIG alignment and text rendering use Qwen2.5-VL-7B-Instruct; reasoning uses LLM2CLIP; style uses CSD and the OneIG StyleEncoder; diversity

uses mean pairwise DreamSim distance. GenEval++ uses GPT-4.1 as described below. Training judges and benchmark evaluators are distinct.

GenEval++ contains 280 prompts across seven compositional task categories, with 40 prompts per category (Ye et al., 2025). Its official protocol uses GPT-4.1 to verify a checklist of object, count, color, position, and size requirements; an image passes only when all required conditions are satisfied. The Overall score is the mean of the seven category accuracies, reported on a 0–100 scale here. We follow this protocol with one image per prompt, using the official system prompt and checklist template with GPT-4.1 at temperature 0.

A.5 Refinement from matched initial images

To separate refinement from differences in initial generation, we fix the prompts, plans, and Round 1 images produced by Baseline and rerun only critique, replanning, and refinement with each 80-update checkpoint (Table 6). All post-trained students exceed Baseline's own refinement on OneIG text rendering (78.02–79.63 versus 73.44) and WISE (66.70–68.20 versus 64.70). On LongText, the two objectives that include OPD reach 68.26–68.74, whereas GRPO reaches 58.38, below Baseline's 60.28. Because the Round 1 inputs are shared, these LongText differences arise in the three later stages; the comparison evaluates these stages jointly and does not attribute the differences to individual stages. No refined output exceeds the common Round 1 WISE score of 68.40.

Table 6: Refinement from matched initial images. All methods share Baseline's initial prompts, plans, and images, and rerun only critique, replanning, and refinement. Baseline reports its own refined outputs.

Refinement stages	OneIG-EN					LongText	WISE
	Align.	Text	Reason.	Style	Div.		
Shared Round 1	85.83	49.52	21.75	43.73	15.72	44.90	68.40
Baseline	83.78	73.44	32.52	44.09	20.84	60.28	64.70
GRPO	84.15	78.02	32.85	45.54	21.55	58.38	66.70
Pure OPD	84.92	79.18	32.78	44.40	19.61	68.74	67.90
GRPO + OPD	85.23	79.63	32.50	44.06	19.24	68.26	68.20

A.6 QUALITATIVE COMPARISON

Figure 4 compares selected outputs from BAGEL, BAGEL with self-CoT, Joint SFT, and ProOPD on identical prompts. Joint SFT and ProOPD show their final outputs after refinement. The examples cover object counting, text rendering, time-zone reasoning, and physical plausibility.

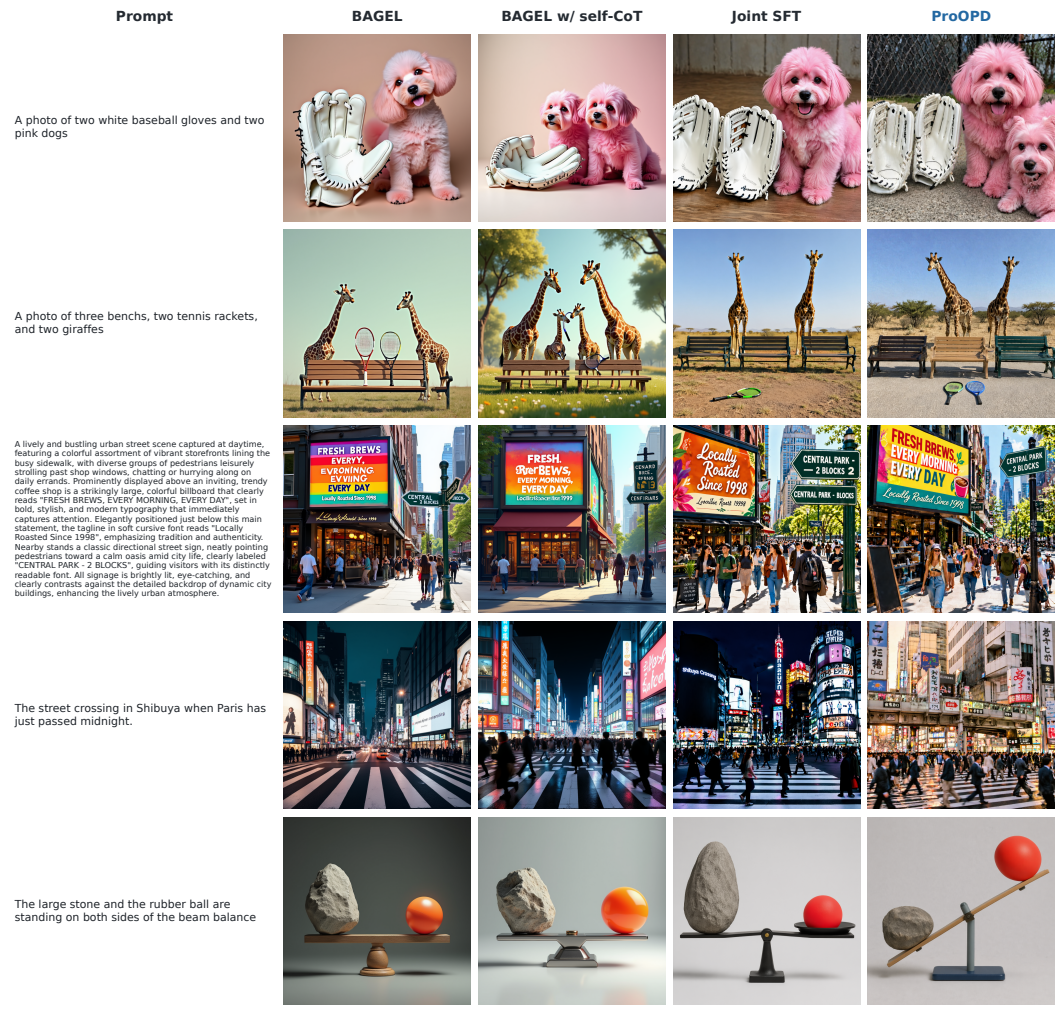

Figure 4: Qualitative comparison on the same prompts. Joint SFT and ProOPD show outputs after refinement.