PERSONALIZING FEW-STEP DIFFUSION DISTILLATION VIA STEPWISE REWARD AUGMENTATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Personalizing a step-distilled diffusion model requires learning a new subject or style while retaining few-step inference. Reference-conditioned on-policy self-distillation (OPSD) provides teacher supervision along the student's sampling trajectory, but encourages reference fidelity and image quality only indirectly. We propose **Stepwise Reward Augmentation (SRA)**, which combines this supervision with local targets constructed from normalized reward gradients at intermediate denoising states. Schedule-derived weights allocate reward supervision across states, while aggregate gradient-norm balancing controls its strength relative to distillation. Training uses detached rollout states without backpropagating through sampling transitions. A gradient interpretation and matched controls clarify the roles of these design choices and inform practical recommendations for allocating reward feedback across sampling states and balancing it with teacher supervision. Experiments on object and style personalization across two backbones demonstrate improved reward-related reference similarity and image quality over teacher-only distillation and uniform-weight reward baselines. Additional reward recipes, new personalization cases, and training-cost comparisons further support the method's applicability and competitive efficiency.

1 INTRODUCTION

Diffusion models have become a powerful paradigm for high-quality image generation (Ho et al., 2020, Song et al., 2021b, Rombach et al., 2022), but their iterative denoising process typically requires many network evaluations, resulting in substantial inference latency and computational cost. To enable more efficient generation, recent work has developed few-step diffusion models that compress the sampling process into only a handful of denoising steps while retaining strong generation quality, through accelerated deterministic sampling (Song et al., 2021a), distribution matching distillation (Yin et al., 2024a,b), and consistency-based generation (Song et al., 2023, Luo et al., 2023).

Among these approaches, step distillation yields models that retain strong generation quality with only a few sampling steps (Yin et al., 2024a, Song et al., 2023, Luo et al., 2023). However, practical applications often require adapting these general-purpose generators to specific subjects, styles, or domains (Chen et al., 2025, Mou et al., 2025, Roy et al., 2025, Kim et al., 2026). Such personalization is particularly challenging for step-distilled models. Adaptation must acquire the desired visual concept while preserving the carefully distilled sampling dynamics that enable high-quality generation in only a few steps (Miao et al., 2025, Jiang et al., 2026). Conventional fine-tuning objectives can introduce a mismatch between the states used for training and those under inference, which may degrade the model's distilled generation capabilities (Yin et al., 2024a, Jiang et al., 2026).

Existing approaches address this challenge through two broad strategies: joint adaptation and distillation from a many-step teacher (Bahram et al., 2026, Yang et al., 2026), and on-policy self-distillation for models that are already step-distilled (Jiang et al., 2026). In the latter setting, D-OPSD (Jiang et al., 2026) proposes on-policy self-distillation (OPSD): the student samples its own few-step trajectories, while a privileged, reference-conditioned EMA teacher provides supervision at the states visited by the student. This on-policy formulation enables the model to acquire new concepts and styles while preserving its original few-step inference capability. However, personalization is learned entirely through agreement with the teacher. Reference fidelity and image quality are therefore only indirectly encouraged through the teacher's predictions rather than optimized explicitly.

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.

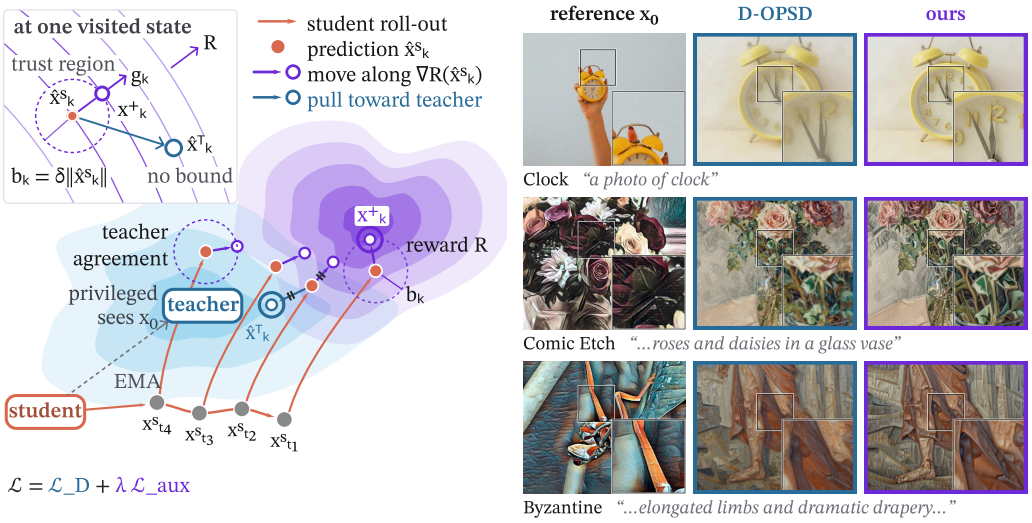


Figure 1: **Stepwise reward augmentation for on-policy self-distillation.** Left: image rewards are evaluated on decoded student predictions along its own few-step rollout. Bounded moves along normalized reward gradients provide local reward supervision. The student combines this reward supervision with reference-conditioned EMA teacher supervision without backpropagating through the rollout. The inset shows the displacement bound on the local reward update. Right: under matched prompts and seeds, SRA produces clearer clock numerals and hands, sharper floral outlines in Comic Etch, and more distinct angular contours and drapery folds in Byzantine than D-OPSD. The 2× crops highlight these details.

To optimize reference fidelity and image quality explicitly, differentiable image rewards provide direct supervision through policy gradients (Black et al., 2024), backpropagation through late-step predictions or sampling trajectories (Xu et al., 2023, Clark et al., 2024), and reward-augmented distillation objectives (Huang et al., 2026a, Jacq et al., 2026). For example, LRR uses learned latent rewards to supervise individual sampling steps (Guan et al., 2026). Reward supervision at different rollout states updates the same student parameters as reference-conditioned teacher supervision.

In this work, we propose Stepwise Reward Augmentation (SRA), which augments reference-conditioned distillation with reward supervision across the student's few-step rollout. At every intermediate denoising state along the student's few-step trajectory, we combine reference-conditioned teacher supervision with local reward using one normalized ascent step on the student's clean prediction. Coefficients from the sampling schedule weight reward contributions across states, while aggregate gradient-norm balancing controls their overall strength relative to distillation. Training uses detached rollout states without backpropagation through sampling transitions. Figure 1 illustrates this joint supervision and the resulting improvements in object detail and style fidelity.

Our experiments on object and style personalization across two backbones demonstrate that SRA improves reward-related reference similarity and image quality over teacher-only distillation and uniform-weight reward baselines. We provide a gradient interpretation of SRA and use matched controls to explain the roles of its design choices. These analyses provide practical guidance for allocating reward feedback across sampling states and balancing it with teacher supervision. Our contributions are as follows:

- We introduce SRA, a reward-augmented on-policy distillation method for reference-conditioned personalization of already step-distilled models. It jointly fits teacher predictions and local targets constructed from normalized reward gradients, with schedule-derived reward weights and explicit control of aggregate reward strength, without backpropagating through sampling transitions.
- We provide supported guidance for reward supervision across few-step sampling states. By combining a gradient interpretation of SRA with matched controls, we explain the roles of its design choices and derive practical recommendations for allocating reward feedback and balancing it with teacher supervision.

- Experiments on object and style personalization across two backbones demonstrate improved reward-related reference similarity and image quality over teacher-only distillation and uniform-weight reward baselines. Additional reward recipes and new personalization cases on Z-Image-Turbo, together with training-cost comparisons, support the method's applicability and competitive efficiency.

2 PRELIMINARIES

2.1 RELATED WORK

On-policy distillation trains a student on the states it samples itself (Agarwal et al., 2024, Gu et al., 2024), and self-distillation makes the teacher the same model given privileged context (Zhao et al., 2026, Shenfeld et al., 2026). D-OPSD (Jiang et al., 2026) carries this to step-distilled diffusion with an EMA copy that sees the reference image.

Differentiable-reward methods backpropagate through late-step predictions (Xu et al., 2023) or sampling trajectories (Clark et al., 2024, Wang et al., 2026). Policy-gradient approaches use scalar rewards over stochastic trajectories (Liu et al., 2025, Xue et al., 2025), with extensions for temporal credit assignment (He et al., 2026, Li et al., 2026, Hu et al., 2025). MeanFlowNFT (Huang et al., 2026b) extends forward-process reinforcement to average-velocity generators. Best-of-N selection uses image rewards to select complete samples (Ma et al., 2025a).

Reward-guided fitting uses positive and negative samples or targets (Zheng et al., 2026, Zhou et al., 2026) and group preferences in DGPO (Luo et al., 2026). LRR (Guan et al., 2026) learns latent reward registers and uses their gradients for stepwise on-policy target fitting. Adjoint Matching (Domingo-Enrich et al., 2025) constructs regression targets from adjoint equations. RTDMD (Huang et al., 2026a) and RMMD (Jacq et al., 2026) combine reward optimization with distribution matching and moment matching distillation, respectively. Our work combines local reward feedback across visited states with reference-conditioned OPSD for personalizing already step-distilled models.

2.2 PROBLEM SETTING

We consider personalizing a pretrained, step-distilled diffusion model using training pairs $(x_0, y) \sim \mathcal{D}$ of reference images and prompts. Each training sample contains one reference image, which conditions the teacher query. Our experiments use four reference and prompt pairs per object or style case. We consider a learned flow-matching velocity field $v_\theta(x_t, t, c)$ on a latent state $x_t \in \mathbb{R}^d$, $t \in [0, 1]$, under a condition c (Liu et al., 2023, Lipman et al., 2023), where θ denotes the model parameters. We sample the model with K Euler steps over the schedule $1 = t_K > \cdots > t_0 = 0$. The training and deployment schedules are given in Appendix D.1.

We adopt the reference-conditioned on-policy self-distillation formulation of D-OPSD (Jiang et al., 2026). The same model is student and teacher under two conditions from one encoder, $c_s = f_{\text{text}}(y)$ and $c_T = f_{\text{mm}}(y, x_0)$, where f_{text} encodes the prompt and f_{mm} the multimodal context of prompt and reference image. Superscripts s and T mark the student's and the teacher's quantities throughout. With velocity $u_k^s = v_\theta(x_{t_k}^s, t_k, c_s)$, the student rolls out from $x_{t_K}^s \sim \mathcal{N}(0, I)$ using $x_{t_{k-1}}^s = x_{t_k}^s + (t_{k-1} - t_k)u_k^s$ for $k = K, \ldots, 1$. The teacher, with parameters $\bar{\theta}$ an exponential moving average of θ, is conditioned on c_T and predicts $u_k^T = v_{\bar{\theta}}(x_{t_k}^s, t_k, c_T)$ at the same visited states.

The corresponding student and teacher clean-latent predictions are

$$\hat{x}_k^s = x_{t_k}^s - t_k u_k^s, \qquad \hat{x}_k^T = x_{t_k}^s - t_k u_k^T. \tag{1}$$

D-OPSD matches these predictions along the student's rollout:

$$\mathcal{L}_D(\theta) = \mathbb{E}_{(x_0,y)\sim\mathcal{D}} \left[\frac{1}{K} \sum_{k=1}^{K} \left\| \hat{x}_k^s - \text{sg}(\hat{x}_k^T) \right\|_2^2 \right]. \tag{2}$$

Here sg denotes stop-gradient. Expectations implicitly include the initial noise. Rollout states and teacher predictions are held fixed during each student update.

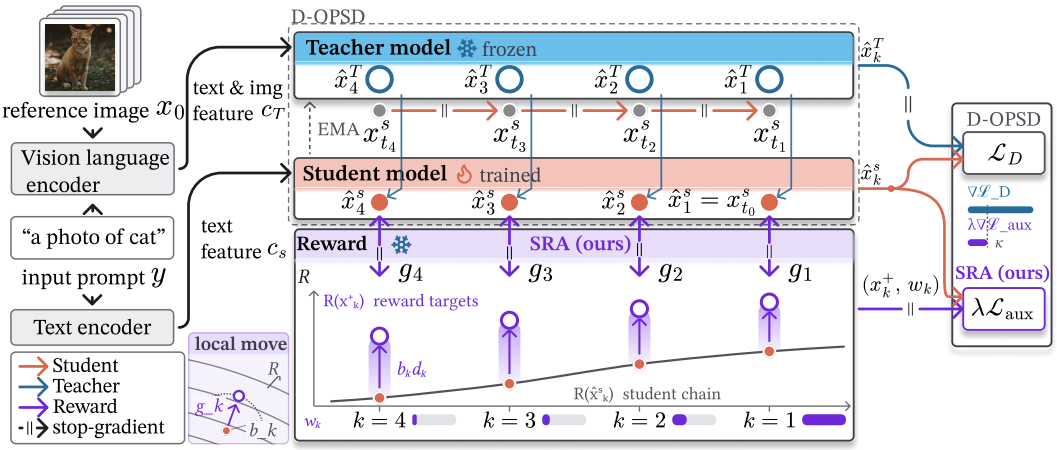


Figure 2: **Stepwise Reward Augmentation.** At states visited by the text-conditioned student, SRA combines local reward with reference-conditioned EMA teacher supervision. Bounded moves along normalized reward gradients provide local reward supervision, and the Euler coefficients c_k weight its contributions across states. The coefficient λ targets a reward-to-distillation raw-gradient norm ratio κ, subject to the numerical safeguards in Appendix D.3. In the diagram, w_k denotes c_k.

In addition to teacher supervision, we assume access to a differentiable reward function $R(x)$ that scores the generated sample x. The reward may depend on the reference image x_0 or the language prompt y. We suppress these arguments when they are fixed or absent (Appendix C). We seek to improve the expected generated-image reward over training pairs and initial noise through joint use of teacher supervision and reward feedback, while retaining the prescribed few-step inference procedure.

3 Stepwise Reward Augmentation

We now augment the distillation objective in Eq. (2) with local reward supervision. With the student parameters fixed, we differentiate R with respect to the student's clean prediction to obtain $g_k = \nabla R(\hat{x}^s_k)$. This derivative passes through the decoder and reward model without propagating through the sampling trajectory (Clark et al., 2024). We take one normalized ascent step using the local reward gradient:

$$x^+_k = \hat{x}^s_k + b_k d_k, \qquad b_k = \delta \|\hat{x}^s_k\|_2, \qquad d_k = g_k / \|g_k\|_2. \tag{3}$$

The hyperparameter δ specifies the relative displacement of this step in prediction space ($d_k = 0$ when $g_k = 0$).

We weight the reward-regression terms across states using the immediate Euler transition coefficient:

$$x^s_{t_{k-1}} = x^s_{t_k} + c_k\big(\hat{x}^s_k - x^s_{t_k}\big), \qquad c_k = \frac{t_k - t_{k-1}}{t_k}. \tag{4}$$

The coefficients c_k are fixed by the sampling schedule but vary across sampling steps. At the final visited state, $c_1 = 1$ and the clean prediction equals the final latent, $x^s_{t_0} = \hat{x}^s_1$. The effect on the final sample also depends on subsequent model responses (Appendix B). At the same fixed rollout states, we combine teacher supervision with reward supervision through regression. The teacher predictions and x^+_k are treated as constants during the student update. For a fixed reward coefficient $\lambda \geq 0$, the joint objective is

$$\mathcal{L}(\theta;\lambda) = \mathcal{L}_D(\theta) + \lambda \underbrace{\mathbb{E}_{(x_0,y)\sim\mathcal{D}}\left[\frac{1}{2K}\sum_{k=1}^{K} c_k \big\|\hat{x}^s_k - \mathrm{sg}(x^+_k)\big\|_2^2\right]}_{\mathcal{L}_{\mathrm{aux}}(\theta)}. \tag{5}$$

We set λ to target a prescribed reward-to-distillation gradient-norm ratio κ, measured using aggregated raw parameter gradients before clipping and AdamW preconditioning (Chen et al., 2018). The coefficient is held fixed during each student update (Appendix D.3).

Algorithm 1 separates local reward evaluation from the student update. Each reward is differentiated with respect to the detached clean prediction, the chain advances from the detached prediction, and parameter gradients flow only through the student predictions at the visited states. The teacher forward at the same states is gradient-free. Each update uses K local reward gradients and two backward passes. Appendix D.2 gives initialization, batch aggregation, the two-pass structure of the implementation, and a recorded endpoint check that the update does not use.

Algorithm 1: One SRA training update

Require: Student LoRA θ (frozen base), EMA teacher $\bar{\theta}$, pair (x_0, y)
Require: Reward R, schedule $\{t_k\}$, δ, κ
Encode c_s (prompt), c_T (prompt, reference) and draw $x^s_{t_K} \sim \mathcal{N}(0, I)$
for $k = K, \ldots, 1$ **do**
  $\hat{x}^T_k \leftarrow$ Eq. (1) with $\bar{\theta}, c_T$ at $x^s_{t_k}$ (no gradient)
  $\hat{x}^s_k \leftarrow$ Eq. (1) with θ, c_s at $x^s_{t_k}$ (with gradient)
  $g_k \leftarrow \nabla_z R(z)|_{z=\mathrm{sg}(\hat{x}^s_k)}$ via decoder, setting non-finite entries to 0
  $x^+_k \leftarrow \mathrm{sg}(\hat{x}^s_k) + \delta \|\hat{x}^s_k\|_2 \, g_k / \|g_k\|_2$ (Eq. (3))
  Advance to $x^s_{t_{k-1}}$ by Eq. (4) from $\mathrm{sg}(\hat{x}^s_k)$
end for
Compute $\widehat{\mathcal{L}}_D, \widehat{\mathcal{L}}_{\mathrm{aux}}$ on the current batch (Eqs. (2), (5))
$G_D \leftarrow \nabla_\theta \widehat{\mathcal{L}}_D$, $G_{\mathrm{aux}} \leftarrow \nabla_\theta \widehat{\mathcal{L}}_{\mathrm{aux}}$
Fix $\lambda \leftarrow \kappa \|G_D\|_2 / \|G_{\mathrm{aux}}\|_2$ (Appendix D.3)
Step θ with AdamW on $G_D + \lambda G_{\mathrm{aux}}$
$\bar{\theta} \leftarrow \mu\bar{\theta} + (1-\mu)\theta$

At fixed parameters and rollout states, using the same local reward gradients, the target-fitting rule rescales each local reward-ascent contribution by the prediction norm divided by the reward-gradient norm. The Euler-weighted ReFL control shares these state weights and aggregate norm balancing, so comparing SRA with Euler-weighted ReFL evaluates the combined local rescaling. The final-state-only variant removes reward supervision at earlier states. The uniform variant assigns every state weight one. Both retain the same local reward update, teacher supervision at all states, and the aggregate norm-balancing rule. Appendix B gives the gradient identity and single-update conditions under which exact norm balancing cancels the common displacement scale δ. The parameter κ controls the raw reward-to-distillation gradient-norm ratio. Tables 2 and 3 compare these design choices and the gradient-norm ratio.

4 Experiments

Experimental setup. Our main validation compares Z-Image-Turbo (Z-Image Team et al., 2025) and FLUX.2-klein-4B (Black Forest Labs, 2025, Black Forest Labs et al., 2025) on ten object and ten style cases, with four references and 1,000 finetuning updates per case. Within the reward-augmented OPSD comparisons, we match the optimizer, rollout and deployment schedules, seed rules, training budget, and reward-to-distillation gradient-norm ratio. Our ReFL adaptation maximizes the reward of each clean prediction at detached rollout states. Our DRaFT adaptation differentiates the final-image reward through the sampling trajectory. Both main-table adaptations use uniform state weights. The main reward combines DINOv3 (Siméoni et al., 2026) reference similarity and TOPIQ (Chen et al., 2024) image quality. DINO-D, DINOv3, and Quality-S share scorer families with this reward. LPIPS-D, VLM-J, CLIP-S, and Aesthetic-S are not used in it. We score fixed final checkpoints on the validation set using four reference rewrites and sixteen evaluation-prompt images per case. DINO-D and LPIPS-D measure correspondence on reference rewrites, whereas DINOv3 measures similarity to the reference set on evaluation-prompt images. Quality-S and Aesthetic-S assess images without prompts. Appendices D, E, and E.3 specify the training and deployment schedules, metric definitions, and baseline adaptations.

Main results. Table 1 shows that SRA achieves the lowest mean DINO-D and the highest mean DINOv3 and Quality-S on both backbones. These gains over teacher-only D-OPSD and the uniform-weight ReFL and DRaFT adaptations support improved reward-related reference similarity and image quality within the shared OPSD protocol. Figure 4 illustrates stronger retention of surface structure and fine detail under matched prompts and seeds. Mean VLM-J and Aesthetic-S also improve over D-OPSD, while LPIPS-D remains comparable and CLIP-S is lower. Thus, the reward-related gains are accompanied by trade-offs on other metrics. We next examine how local reward construction, state allocation, and aggregate strength contribute to these results.

Reward supervision design. Table 2 varies the reward supervision on Z-Image-Turbo at the aggregate gradient-norm ratio of SRA, retaining teacher supervision except in the last row.

Table 1: **Deployed student on the validation set.** Mean performance over ten object and ten style cases for each base model. Subscripts give the change from D-OPSD on the same base model. Green / red mark better / worse in the direction of each column's arrow. Bold and underline denote the best and second-best values within each model. ReFL and DRaFT denote reward-augmented OPSD adaptations with uniform state weights. DINO-D, DINOv3, and Quality-S share scorer families with the training reward. Appendix E specifies baselines, metric definitions, evaluation panels, and scorer qualifications.

Method	DINO-D↓	LPIPS-D↓	VLM-J↑	CLIP-S↑	DINOv3↑	Quality-S↑	Aesthetic-S↑
Z-Image-Turbo							
Base model	$0.3002_{\uparrow 0.0724}$	$0.7723_{\uparrow 0.0507}$	$1.0000_{\downarrow 0.4531}$	$\mathbf{0.3308}_{\uparrow 0.0081}$	$0.6348_{\downarrow 0.0444}$	$0.6301_{\downarrow 0.0107}$	$5.4046_{\uparrow 0.1015}$
SFT	$0.2637_{\uparrow 0.0359}$	$\mathbf{0.6772}_{\downarrow 0.0444}$	$1.5094_{\uparrow 0.0563}$	$\underline{0.3263}_{\uparrow 0.0036}$	$0.6214_{\downarrow 0.0578}$	$0.3328_{\downarrow 0.3080}$	$4.7807_{\downarrow 0.5224}$
DreamBooth	$0.2641_{\uparrow 0.0363}$	$\underline{0.6947}_{\downarrow 0.0269}$	$1.4125_{\downarrow 0.0406}$	$0.3257_{\uparrow 0.0030}$	$0.6399_{\downarrow 0.0393}$	$0.3746_{\downarrow 0.2662}$	$4.8767_{\downarrow 0.4264}$
PSO	$0.2941_{\uparrow 0.0663}$	$0.7567_{\uparrow 0.0351}$	$1.0375_{\downarrow 0.4156}$	$0.3258_{\uparrow 0.0031}$	$0.6325_{\downarrow 0.0467}$	$0.4308_{\downarrow 0.2100}$	$5.0493_{\downarrow 0.2538}$
SFT → LoRA	$0.2105_{\downarrow 0.0173}$	$0.7260_{\uparrow 0.0044}$	$\underline{1.6938}_{\uparrow 0.2407}$	$0.3182_{\downarrow 0.0045}$	$0.6918_{\uparrow 0.0126}$	$0.5643_{\downarrow 0.0765}$	$5.2118_{\downarrow 0.0913}$
D-OPSD	0.2278	0.7216	1.4531	0.3227	0.6792	0.6408	5.3031
+ ReFL	$0.1825_{\downarrow 0.0453}$	$0.7180_{\downarrow 0.0036}$	$1.6219_{\uparrow 0.1687}$	$0.3196_{\downarrow 0.0031}$	$0.7019_{\uparrow 0.0227}$	$0.7471_{\uparrow 0.1062}$	$5.5386_{\uparrow 0.2355}$
+ DRaFT	$\underline{0.1804}_{\downarrow 0.0474}$	$0.7217_{\uparrow 0.0001}$	$1.6313_{\uparrow 0.1782}$	$0.3201_{\downarrow 0.0026}$	$\underline{0.7063}_{\uparrow 0.0271}$	$\underline{0.7550}_{\uparrow 0.1142}$	$\underline{5.5721}_{\uparrow 0.2690}$
+ SRA	$\mathbf{0.1503}_{\downarrow 0.0775}$	$0.7218_{\uparrow 0.0002}$	$\mathbf{1.7625}_{\uparrow 0.3094}$	$0.3191_{\downarrow 0.0036}$	$\mathbf{0.7283}_{\uparrow 0.0491}$	$\mathbf{0.8382}_{\uparrow 0.1973}$	$\mathbf{5.6285}_{\uparrow 0.3254}$
FLUX.2-klein-4B							
Base model	$0.2878_{\uparrow 0.0924}$	$0.7686_{\uparrow 0.0444}$	$1.0562_{\downarrow 0.7313}$	$\mathbf{0.3535}_{\uparrow 0.0181}$	$0.6291_{\downarrow 0.0916}$	$0.5834_{\uparrow 0.0528}$	$5.5681_{\uparrow 0.3168}$
SFT	$0.2447_{\uparrow 0.0493}$	$\underline{0.6831}_{\downarrow 0.0411}$	$1.3844_{\downarrow 0.4031}$	$0.3352_{\downarrow 0.0002}$	$0.6690_{\downarrow 0.0517}$	$0.3408_{\downarrow 0.1898}$	$5.1598_{\downarrow 0.0915}$
DreamBooth	$0.2273_{\uparrow 0.0319}$	$\mathbf{0.6805}_{\downarrow 0.0437}$	$1.7062_{\downarrow 0.0813}$	$0.3321_{\downarrow 0.0033}$	$0.7078_{\downarrow 0.0129}$	$0.3606_{\downarrow 0.1700}$	$5.2049_{\downarrow 0.0464}$
PSO	$0.3748_{\uparrow 0.1794}$	$0.7855_{\uparrow 0.0613}$	$1.1125_{\downarrow 0.6750}$	$0.3408_{\uparrow 0.0054}$	$0.6272_{\downarrow 0.0935}$	$0.4902_{\downarrow 0.0404}$	$5.0443_{\downarrow 0.2070}$
SFT → LoRA	$0.2203_{\uparrow 0.0249}$	$0.7219_{\downarrow 0.0023}$	$1.6938_{\downarrow 0.0937}$	$\underline{0.3471}_{\uparrow 0.0117}$	$0.6718_{\downarrow 0.0489}$	$0.5005_{\downarrow 0.0301}$	$5.2132_{\downarrow 0.0381}$
D-OPSD	0.1954	0.7242	1.7875	0.3354	0.7207	0.5306	5.2513
+ ReFL	$\underline{0.1650}_{\downarrow 0.0304}$	$0.7224_{\downarrow 0.0018}$	$1.9188_{\uparrow 0.1313}$	$0.3327_{\downarrow 0.0027}$	$\underline{0.7343}_{\uparrow 0.0136}$	$\underline{0.7232}_{\uparrow 0.1926}$	$\underline{5.5737}_{\uparrow 0.3224}$
+ DRaFT	$0.1654_{\downarrow 0.0300}$	$0.7224_{\downarrow 0.0018}$	$\underline{2.0219}_{\uparrow 0.2344}$	$0.3343_{\downarrow 0.0011}$	$0.7330_{\uparrow 0.0123}$	$0.7107_{\uparrow 0.1801}$	$5.5528_{\uparrow 0.3015}$
+ SRA	$\mathbf{0.1417}_{\downarrow 0.0538}$	$0.7200_{\downarrow 0.0042}$	$\mathbf{2.1250}_{\uparrow 0.3375}$	$0.3316_{\downarrow 0.0038}$	$\mathbf{0.7469}_{\uparrow 0.0262}$	$\mathbf{0.7793}_{\uparrow 0.2487}$	$\mathbf{5.6057}_{\uparrow 0.3544}$

Local reward construction. At fixed Euler weights, SRA improves all three reward-related means over direct ReFL updates and over targets built from the reward backpropagated through the remaining steps. Its advantage over ReFL holds at every tested κ (Table 3). Euler-weighted DRaFT reaches similar DINOv3 and Quality-S means but worse DINO-D and LPIPS-D. The two-sided construction, which regresses toward half the difference between reward-ascent and reward-descent proposals, improves the three reward-related means further but requires three reward-gradient evaluations per state instead of one. The remaining target rows weight backpropagated, random, or two-sided targets by their measured endpoint reward gains, or split the backpropagated step into two half-steps. None exceeds SRA on all three metrics, and random directions stay near D-OPSD. The one-sided local construction thus balances reward-related performance against reward computation (Appendix E.4).

State allocation. Euler weights improve the three reward-related means over uniform weights within the ReFL, one-sided, and two-sided constructions. The last-heavy weights $[0.2, 0.2, 0.2, 1]$ come close to SRA, with higher DINOv3 and Quality-S but slightly worse DINO-D, so the gain comes from emphasizing later states, with Euler coefficients as a default tied to the sampling transitions. Supervising only the final state raises DINOv3 and Quality-S on evaluation-prompt images but worsens DINO-D and LPIPS-D on reference rewrites, so earlier states still need reward supervision for rewrite correspondence.

Teacher supervision and reward strength. Table 3 compares SRA with Euler-weighted ReFL on Z-Image-Turbo at $\kappa \in \{0.1, 0.3, 0.6\}$, matching the state weights and prescribed reward-to-distillation gradient-norm ratio. Increasing κ improves the three reward-related metrics for both constructions. The *No teacher* variant in Table 2, which removes distillation and its gradient-norm balancing, raises the reward-related means further but worsens all four complementary metrics. Reward strength should therefore be set relative to teacher supervision, with κ chosen for the desired balance between reward gains and other personalization objectives.

Training dynamics and cost. Figure 3 evaluates training dynamics and computational cost on Z-Image-Turbo. SRA reaches the highest final training reward in both case groups and maintains the highest mean validation Quality-S across the measured checkpoints. On one H200, its mean cost is 2.12 GPU hours per 1,000 updates, close to uniform-weight ReFL's 2.05 and below DRaFT's 3.26. SRA also matches the final metric means of each compared method earlier. On the 50-update

Table 2: **Reward supervision ablations.** Z-Image-Turbo, twenty validation cases, one training run per configuration and case. Subscripts give changes from SRA (purple). Green / red mark better / worse means, without indicating statistical significance. Uniform-weight ReFL repeats Table 1. Uniform-weight DRaFT is reported there. Appendix E.4 gives configuration definitions and paired effects.

	Reward-related metrics			Other evaluation metrics			
Configuration	DINO-D↓	DINOv3↑	Quality-S↑	LPIPS-D↓	VLM-J↑	CLIP-S↑	Aesthetic-S↑
Reward construction and state weighting							
ReFL, uniform	$0.1825_{\uparrow 0.0323}$	$0.7019_{\downarrow 0.0264}$	$0.7471_{\downarrow 0.0911}$	$0.7180_{\downarrow 0.0038}$	$1.6219_{\downarrow 0.1406}$	$0.3196_{\uparrow 0.0005}$	$5.5386_{\downarrow 0.0899}$
ReFL, Euler	$0.1591_{\uparrow 0.0088}$	$0.7229_{\downarrow 0.0054}$	$0.8113_{\downarrow 0.0269}$	$0.7197_{\downarrow 0.0022}$	$1.7375_{\downarrow 0.0250}$	$0.3182_{\downarrow 0.0009}$	$5.6278_{\downarrow 0.0006}$
One-sided, uniform	$0.1686_{\uparrow 0.0184}$	$0.7096_{\downarrow 0.0187}$	$0.7831_{\downarrow 0.0551}$	$0.7195_{\downarrow 0.0023}$	$1.6594_{\downarrow 0.1031}$	$0.3194_{\uparrow 0.0003}$	$5.6025_{\downarrow 0.0260}$
One-sided, Euler (**SRA**)	0.1503	0.7283	0.8382	0.7218	1.7625	0.3191	5.6285
One-sided, last-heavy	$0.1520_{\uparrow 0.0018}$	$0.7309_{\uparrow 0.0026}$	$0.8437_{\uparrow 0.0056}$	$0.7230_{\uparrow 0.0011}$	$1.7062_{\downarrow 0.0562}$	$0.3189_{\downarrow 0.0002}$	$5.6420_{\uparrow 0.0135}$
Two-sided, uniform	$0.1695_{\uparrow 0.0193}$	$0.7150_{\downarrow 0.0133}$	$0.8012_{\downarrow 0.0369}$	$0.7197_{\downarrow 0.0022}$	$1.6656_{\downarrow 0.0969}$	$0.3210_{\uparrow 0.0019}$	$5.6407_{\uparrow 0.0123}$
Two-sided, Euler	$0.1474_{\downarrow 0.0029}$	$0.7336_{\uparrow 0.0053}$	$0.8593_{\uparrow 0.0212}$	$0.7237_{\uparrow 0.0019}$	$1.6875_{\downarrow 0.0750}$	$0.3190_{\downarrow 0.0001}$	$5.6449_{\uparrow 0.0164}$
DRaFT, Euler	$0.1593_{\uparrow 0.0090}$	$0.7270_{\downarrow 0.0013}$	$0.8325_{\downarrow 0.0057}$	$0.7257_{\uparrow 0.0039}$	$1.8156_{\uparrow 0.0531}$	$0.3188_{\downarrow 0.0003}$	$5.6654_{\uparrow 0.0369}$
State coverage							
Final state only	$0.1554_{\uparrow 0.0051}$	$0.7336_{\uparrow 0.0053}$	$0.8479_{\uparrow 0.0097}$	$0.7296_{\uparrow 0.0078}$	$1.7594_{\downarrow 0.0031}$	$0.3182_{\downarrow 0.0009}$	$5.6143_{\downarrow 0.0142}$
Gain-weighted targets							
Backpropagated	$0.1781_{\uparrow 0.0278}$	$0.7051_{\downarrow 0.0232}$	$0.7758_{\downarrow 0.0624}$	$0.7214_{\downarrow 0.0004}$	$1.6125_{\downarrow 0.1500}$	$0.3207_{\uparrow 0.0016}$	$5.6466_{\uparrow 0.0181}$
Random	$0.2297_{\uparrow 0.0794}$	$0.6767_{\downarrow 0.0516}$	$0.6402_{\downarrow 0.1980}$	$0.7221_{\uparrow 0.0003}$	$1.4656_{\downarrow 0.2969}$	$0.3213_{\uparrow 0.0022}$	$5.2987_{\downarrow 0.3298}$
Two-sided	$0.1514_{\uparrow 0.0011}$	$0.7333_{\uparrow 0.0050}$	$0.8682_{\uparrow 0.0300}$	$0.7284_{\uparrow 0.0066}$	$1.6938_{\downarrow 0.0687}$	$0.3193_{\uparrow 0.0002}$	$5.5925_{\downarrow 0.0360}$
Backpropagated targets							
Euler weights	$0.1670_{\uparrow 0.0167}$	$0.7195_{\downarrow 0.0088}$	$0.8044_{\downarrow 0.0338}$	$0.7233_{\uparrow 0.0015}$	$1.6719_{\downarrow 0.0906}$	$0.3197_{\uparrow 0.0006}$	$5.6721_{\uparrow 0.0436}$
Two half-steps, gain weights	$0.1757_{\uparrow 0.0254}$	$0.7061_{\downarrow 0.0222}$	$0.7954_{\downarrow 0.0428}$	$0.7229_{\uparrow 0.0011}$	$1.5688_{\downarrow 0.1937}$	$0.3210_{\uparrow 0.0019}$	$5.6568_{\uparrow 0.0283}$
Teacher supervision							
No teacher	$0.1272_{\downarrow 0.0231}$	$0.7621_{\uparrow 0.0338}$	$0.9539_{\uparrow 0.1157}$	$0.7764_{\uparrow 0.0546}$	$1.4219_{\downarrow 0.3406}$	$0.2884_{\downarrow 0.0307}$	$5.5109_{\downarrow 0.1176}$

Table 3: **Matched reward strength.** Means over twenty Z-Image-Turbo cases with shared Euler weights and teacher supervision. $\kappa = 0$: D-OPSD. Better means within each pair in bold.

κ	Method	DINO-D↓	DINOv3↑	Qual-S↑
0	D-OPSD	0.2278	0.6792	0.6408
0.1	ReFL, Euler	0.1827	0.7019	0.7568
	SRA	**0.1783**	**0.7058**	**0.7738**
0.3	ReFL, Euler	0.1591	0.7229	0.8113
	SRA	**0.1503**	**0.7283**	**0.8382**
0.6	ReFL, Euler	0.1402	0.7340	0.8472
	SRA	**0.1351**	**0.7455**	**0.8719**

Table 4: **Additional cases.** Z-Image-Turbo, four objects and four styles. One seed per method and case. ReFL and DRaFT use uniform weights. Best in bold.

Method	DINO-D↓	DINOv3↑	Quality-S↑
D-OPSD	0.1949	0.6968	0.6672
ReFL	0.1654	0.7270	0.7529
DRaFT	0.1638	0.7265	0.7661
SRA	**0.1313**	**0.7499**	**0.8386**

checkpoint grid, it first reaches all three reward-related means of both uniform-weight ReFL and DRaFT at update 400 (0.85 hours), and those of the uniform-weight two-sided construction at update 550 (1.17 hours). SRA thus achieves comparable cost per update while reaching these performance levels sooner. Times exclude scoring. Appendix H gives the full curves and time accounting.

Additional cases. Table 4 evaluates four additional object and four additional style cases on Z-Image-Turbo. We fix the method and hyperparameters before evaluating these test cases, and separately adapt each method to each case using four references and 1,000 updates. SRA achieves the best mean DINO-D, DINOv3, and Quality-S among the compared methods. These results extend the reward-related gains beyond the cases used during development.

Additional reward recipes. Table 5 evaluates five recipes on Z-Image-Turbo: the main DINOv3 and TOPIQ combination, DINOv3 with PickScore, and DINOv3, TOPIQ, or ImageReward alone. Across all five recipes, SRA achieves the best mean performance on the corresponding reward-related metrics among the compared configurations. The gains therefore extend to alternative reward combinations and individual reward objectives under the same OPSD protocol. Over three training seeds, the main-table lead holds on both backbones, with standard deviations of at most 0.004 against a smallest lead of 0.011 (Table 6).

Human preference. Figure 5 provides complementary evidence from comparisons with both references and prompts visible. The case-weighted preference for SRA ranges from 68% against

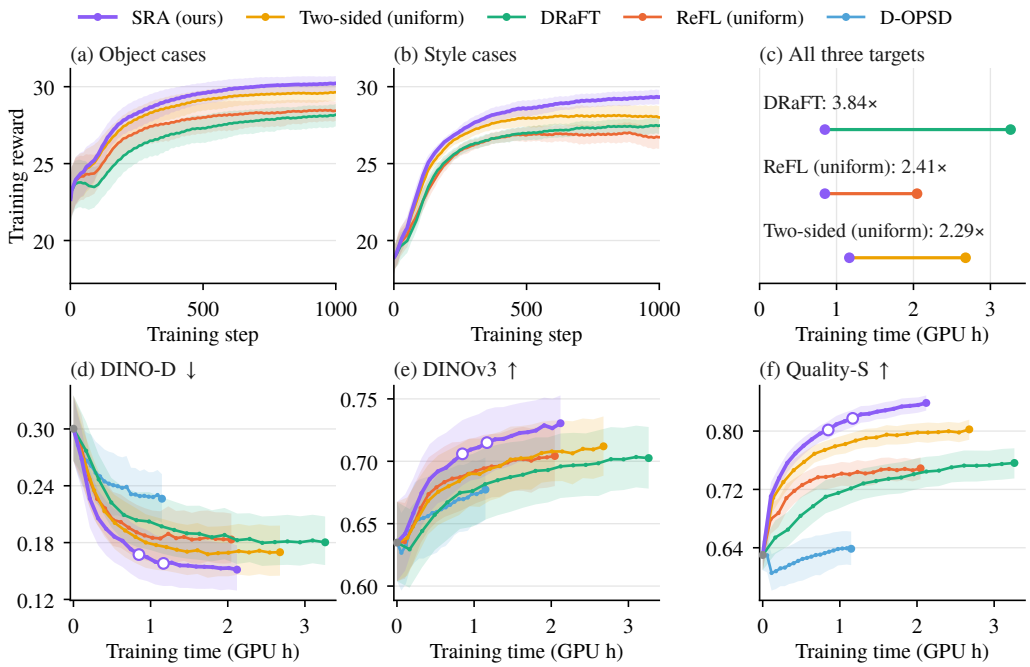


Figure 3: **Training dynamics and efficiency.** (a, b) Mean training-chain reward over ten object and ten style cases (trailing 50-step mean). (c) Time to all three comparator-final metric means: colored dots mark comparators at 1,000 updates, purple dots the first SRA checkpoint reaching all three. Labels give comparator/SRA time ratios. (d, e, f) Validation means over twenty cases against mean training time on one H200, using the time accounting of Appendix D.2. Rings mark the SRA checkpoints in (c). Shading: 95% case bootstrap. Checkpoints are scored every 50 updates. Times exclude scoring. ReFL and the two-sided construction use uniform weights here. Grey dots mark the unadapted base model.

Table 5: **Reward recipes on Z-Image-Turbo.** Each column group uses a different reward and reports only the rewarded metrics, as means over twenty cases. Best in bold. Details in Appendix C.1.

	DINOv3 + TOPIQ (main)			DINOv3 + PickScore			DINOv3 only		TOPIQ only	ImageReward only
Method	DINO-D↓	DINOv3↑	Quality-S↑	DINO-D↓	DINOv3↑	PickScore↑	DINO-D↓	DINOv3↑	Quality-S↑	ImageReward↑
D-OPSD	0.2278	0.6792	0.6408	0.2278	0.6792	22.3596	0.2278	0.6792	0.6408	1.0018
+ ReFL (uniform)	0.1825	0.7019	0.7471	0.1688	0.7180	22.7695	0.1626	0.7189	0.7640	1.2792
+ DRaFT	0.1804	0.7063	0.7550	0.1736	0.7039	22.7158	0.1690	0.7077	0.8250	1.2477
+ SRA	**0.1503**	**0.7283**	**0.8382**	**0.1401**	**0.7323**	**23.2008**	**0.1314**	**0.7494**	**0.8875**	**1.3670**

Table 6: **Three-seed results on both base models.** Mean ± sample standard deviation across three training seeds, with twenty validation cases per model. ReFL and DRaFT use uniform weights. Best mean within each model in bold.

	Z-Image-Turbo			FLUX.2-klein-4B		
Method	DINO-D↓	DINOv3↑	Quality-S↑	DINO-D↓	DINOv3↑	Quality-S↑
D-OPSD	0.2280 ± 0.0005	0.6784 ± 0.0021	0.6431 ± 0.0020	0.1929 ± 0.0034	0.7209 ± 0.0008	0.5291 ± 0.0015
ReFL (uniform)	0.1822 ± 0.0022	0.7040 ± 0.0029	0.7488 ± 0.0017	0.1616 ± 0.0037	0.7362 ± 0.0018	0.7231 ± 0.0006
DRaFT	0.1808 ± 0.0004	0.7064 ± 0.0010	0.7564 ± 0.0014	0.1621 ± 0.0038	0.7336 ± 0.0018	0.7083 ± 0.0025
SRA	**0.1504** ± 0.0015	**0.7284** ± 0.0036	**0.8368** ± 0.0014	**0.1372** ± 0.0040	**0.7471** ± 0.0002	**0.7791** ± 0.0004

DRaFT on FLUX.2-klein-4B to 86% against D-OPSD on Z-Image-Turbo. Seven participants provided 440 judgments covering 275 of the 320 image pairs and all twenty cases in each comparison. Candidates use the main-table checkpoints and matched prompts and seeds, with unlabeled methods and randomized positions. These judgments indicate an overall preference for SRA when references and prompts are visible, while the single question does not separately measure reference fidelity, prompt adherence, or image quality. PickScore and ImageReward evaluations without reference images instead favor D-OPSD and the uniform-weight reward baselines (Appendix F). The automated scores and human judgments thus provide complementary views under different evaluation conditions.

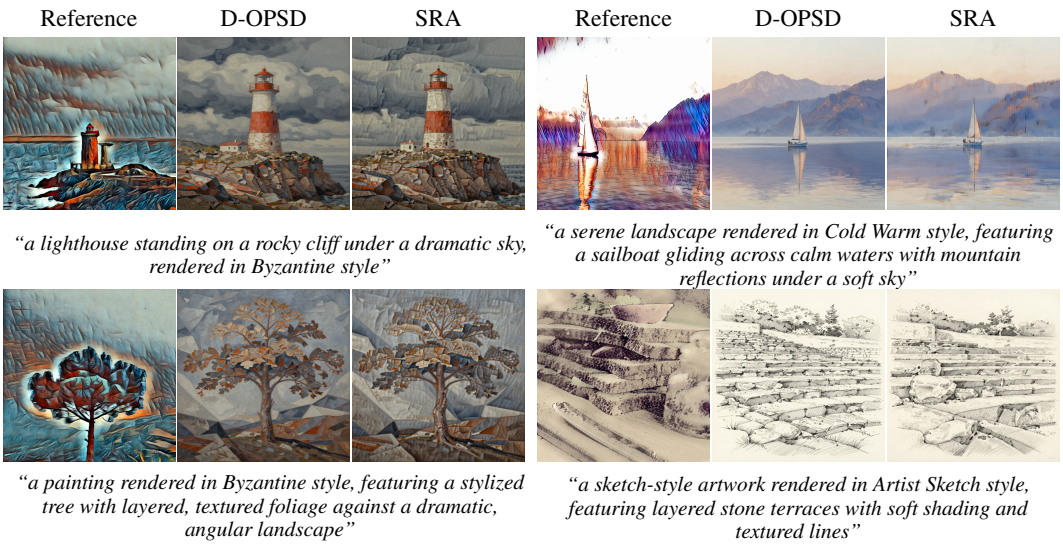


Figure 4: **Qualitative comparison against D-OPSD.** Reference rewrites use matched prompts and seeds. Shown cases favor SRA on DINO-D for all four rewrites. SRA retains more of the reference's surface structure, including relief textures and hand-drawn hatching.

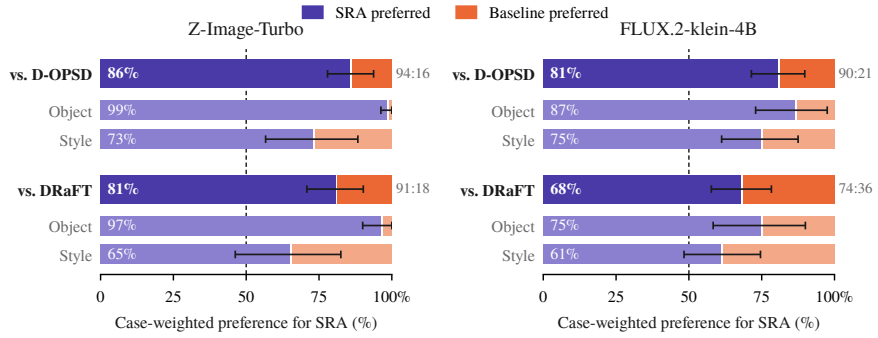


Figure 5: **Human preference.** Case-weighted share of judgments preferring SRA over each baseline: all twenty cases (dark, with the counts for and against SRA) and the object and style groups (light). Whiskers are 95% case bootstraps. Seven participants provided 440 judgments. Appendix G describes the study design.

5 Discussion

SRA combines normalized local reward targets, schedule-derived state weights, and aggregate gradient-norm balancing with reference-conditioned teacher supervision for personalizing step-distilled diffusion models. A gradient interpretation and matched controls decompose the roles of these components, connecting each design choice to its empirical effect and supporting practical guidance for state weighting and reward strength. Across two backbones, SRA improves reward-related reference similarity and image quality over teacher-only distillation and uniform-weight reward baselines, with consistent gains across training seeds and a supplementary human preference study. The current evidence covers reference-conditioned few-step OPSD with differentiable rewards. Broader reward combinations, adaptive state weighting, and scaling to larger case sets remain directions for future work.

AI USE STATEMENT

We used generative AI tools in the following capacities during this work. **Writing.** An LLM assistant was used to polish the manuscript text. All final wording reflects the authors' judgment and review. **Literature discovery.** We periodically queried an LLM with web-search capabilities to scan recent related work. All cited references were verified by the authors. **Code development.** An LLM coding assistant was used to implement the proposed method and baselines, experiment infrastructure, and analysis scripts. AI-generated code was manually reviewed by the authors to ensure correctness before use in reported experiments. **Synthetic data.** Style-case training and evaluation prompts were generated by Qwen3-VL-8B-Instruct (Bai et al., 2025) through the prompt generator and output validator of the D-OPSD reference implementation (Appendix E). We did not use generative AI to propose or refine hypotheses, design or give feedback on research methodology or experiments, interpret results, develop theoretical models, or formulate or prove mathematical claims. We take responsibility for the final content of this work, including all text, claims, and artifacts produced with the aid of generative AI.

ETHICS STATEMENT

All personalization cases come from two public datasets, DreamBooth (Ruiz et al., 2023) and UnlearnCanvas (Zhang et al., 2024), and contain no images of people. The human preference study (Appendix G) asked participants to compare generated images of objects and styles through anonymous links. Participation was voluntary and unpaid, and no personally identifying information was collected. As with other personalization methods, SRA could be misused to reproduce a specific subject or an artist's style without permission. Users should respect the rights attached to their reference images.

REPRODUCIBILITY STATEMENT

Algorithm 1 specifies a complete training update. Appendix C defines the reward components and their calibrated scales. Appendix D lists all hyperparameters (Table S4), training and deployment schedules (Table S3), the gradient-norm balancing rule and its numerical safeguards, and per-method training costs. Appendix E describes the validation cases, metric definitions (Table S5), evaluation panels, and prompt sources. Appendix E.3 gives the baseline configurations, and Appendix E.4 specifies every ablation variant. Training-seed replication across three seeds is reported in Appendix E.1, and test-case evaluation on eight new cases in Appendix E.2. The code for training and evaluation will be made publicly available upon publication.

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 (ICLR)*, 2024. URL https://openreview.net/forum?id=3zKtaqxLhW.

Yara Bahram, Mélodie Desbos, Mohammadhadi Shateri, and Eric Granger. Uni-DAD: Unified distillation and adaptation of diffusion models for few-step few-shot image generation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 26635–26645, 2026. URL https://openaccess.thecvf.com/content/CVPR2026/html/Bahram_Uni-DAD_Unified_Distillation_and_Adaptation_of_Diffusion_Models_for_Few-step_CVPR_2026_paper.html.

Shuai Bai, Yuxuan Cai, Ruizhe Chen, Keqin Chen, Xionghui Chen, Zesen Cheng, Lianghao Deng, Wei Ding, Chang Gao, Chunjiang Ge, Wenbin Ge, Zhifang Guo, Qidong Huang, Jie Huang, Fei Huang, Binyuan Hui, Shutong Jiang, Zhaohai Li, Mingsheng Li, Mei Li, Kaixin Li, Zicheng Lin, Junyang Lin, Xuejing Liu, Jiawei Liu, Chenglong Liu, Yang Liu, Dayiheng Liu, Shixuan Liu, Dunjie Lu, Ruilin Luo, Chenxu Lv, Rui Men, Lingchen Meng, Xuancheng Ren, Xingzhang Ren, Sibo Song, Yuchong Sun, Jun Tang, Jianhong Tu, Jianqiang Wan, Peng Wang, Pengfei Wang, Qiuyue Wang, Yuxuan Wang, Tianbao Xie, Yiheng Xu, Haiyang Xu, Jin Xu, Zhibo Yang, Mingkun Yang, Jianxin Yang, An Yang, Bowen Yu, Fei Zhang, Hang Zhang, Xi Zhang, Bo Zheng, Humen

Zhong, Jingren Zhou, Fan Zhou, Jing Zhou, Yuanzhi Zhu, and Ke Zhu. Qwen3-VL technical report. *arXiv preprint arXiv:2511.21631*, 2025. URL https://arxiv.org/abs/2511.21631.

Kevin Black, Michael Janner, Yilun Du, Ilya Kostrikov, and Sergey Levine. Training diffusion models with reinforcement learning. In *International Conference on Learning Representations (ICLR)*, 2024. URL https://openreview.net/forum?id=YCWjhGrJFD.

Black Forest Labs. FLUX.2: Frontier visual intelligence. Black Forest Labs Blog, 2025. URL https://bfl.ai/blog/flux-2.

Black Forest Labs, Stephen Batifol, Andreas Blattmann, Frederic Boesel, Saksham Consul, Cyril Diagne, Tim Dockhorn, Jack English, Zion English, Patrick Esser, Sumith Kulal, Kyle Lacey, Yam Levi, Cheng Li, Dominik Lorenz, Jonas Müller, Dustin Podell, Robin Rombach, Harry Saini, Axel Sauer, and Luke Smith. FLUX.1 Kontext: Flow matching for in-context image generation and editing in latent space. *arXiv preprint arXiv:2506.15742*, 2025. URL https://arxiv.org/abs/2506.15742.

Chaofeng Chen, Jiadi Mo, Jingwen Hou, Haoning Wu, Liang Liao, Wenxiu Sun, Qiong Yan, and Weisi Lin. TOPIQ: A top-down approach from semantics to distortions for image quality assessment. *IEEE Transactions on Image Processing (TIP)*, 33:2404–2418, 2024. doi: 10.1109/TIP.2024.3378466. URL https://ieeexplore.ieee.org/document/10478301.

Shangyu Chen, Zizheng Pan, Jianfei Cai, and Dinh Phung. PaRa: Personalizing text-to-image diffusion via parameter rank reduction. In *International Conference on Learning Representations (ICLR)*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/f09e8dd9274cb7c2dd0dc65ffc6f427a-Abstract-Conference.html.

Zhao Chen, Vijay Badrinarayanan, Chen-Yu Lee, and Andrew Rabinovich. GradNorm: Gradient normalization for adaptive loss balancing in deep multitask networks. In *International Conference on Machine Learning (ICML)*, pp. 794–803, 2018. URL https://proceedings.mlr.press/v80/chen18a.html.

Kevin Clark, Paul Vicol, Kevin Swersky, and David J. Fleet. Directly fine-tuning diffusion models on differentiable rewards. In *International Conference on Learning Representations (ICLR)*, 2024. URL https://openreview.net/forum?id=1vmSEVL19f.

Carles Domingo-Enrich, Michal Drozdzal, Brian Karrer, and Ricky T. Q. Chen. Adjoint Matching: Fine-tuning flow and diffusion generative models with memoryless stochastic optimal control. In *International Conference on Learning Representations (ICLR)*, 2025. URL https://openreview.net/forum?id=xQBRrtQM8u.

Alex Fang, Albin Madappally Jose, Amit Jain, Ludwig Schmidt, Alexander Toshev, and Vaishaal Shankar. Data filtering networks. In *International Conference on Learning Representations (ICLR)*, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/hash/9bf0810a4a1597a36d27ceea58667d92-Abstract-Conference.html.

Yuxian Gu, Li Dong, Furu Wei, and Minlie Huang. MiniLLM: Knowledge distillation of large language models. In *International Conference on Learning Representations (ICLR)*, 2024. URL https://openreview.net/forum?id=5h0qf7IBZZ.

Yuanshen Guan, Zipeng Feng, Chengru Song, Zhiwei Xiong, and Peiqin Sun. Latent reward registers for diffusion preference alignment. *arXiv preprint arXiv:2608.03929*, 2026. URL https://arxiv.org/abs/2608.03929.

Xiaoxuan He, Siming Fu, Yuke Zhao, Wanli Li, Jian Yang, Dacheng Yin, Fengyun Rao, and Bo Zhang. TempFlow-GRPO: When timing matters for GRPO in flow models. In *International Conference on Learning Representations (ICLR)*, 2026. URL https://openreview.net/forum?id=7mCo3R3Wyn.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 33, 2020. URL https://proceedings.neurips.cc/paper/2020/hash/4c5bcfec8584af0d967f1ab10179ca4b-Abstract.html.

Vlad Hosu, Hanhe Lin, Tamas Sziranyi, and Dietmar Saupe. KonIQ-10k: An ecologically valid database for deep learning of blind image quality assessment. *IEEE Transactions on Image Processing (TIP)*, 29:4041–4056, 2020. doi: 10.1109/TIP.2020.2967829. URL https://doi.org/10.1109/TIP.2020.2967829.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations (ICLR)*, 2022. URL https://openreview.net/forum?id=nZeVKeeFYf9.

Zijing Hu, Fengda Zhang, Long Chen, Kun Kuang, Jiahui Li, Kaifeng Gao, Jun Xiao, Xin Wang, and Wenwu Zhu. Towards better alignment: Training diffusion models with reinforcement learning against sparse rewards. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 23604–23614, 2025. URL https://openaccess.thecvf.com/content/CVPR2025/html/Hu_Towards_Better_Alignment_Training_Diffusion_Models_with_Reinforcement_Learning_Against_CVPR_2025_paper.html.

Yushi Huang, Xiangxin Zhou, Ruoyu Wang, Chi Zhang, Jun Zhang, and Tianyu Pang. Reinforcing few-step generators via reward-tilted distribution matching. *arXiv preprint arXiv:2605.26108*, 2026a. URL https://arxiv.org/abs/2605.26108.

Yushi Huang, Xiangxin Zhou, Jun Zhang, Liefeng Bo, and Tianyu Pang. MeanFlowNFT: Bringing forward-process RL to average-velocity generators. *arXiv preprint arXiv:2607.15273*, 2026b. URL https://arxiv.org/abs/2607.15273.

Alexis Jacq, Guillaume Couairon, Valentin De Bortoli, Quentin Berthet, Arnaud Doucet, and Romuald Elie. Diffusion fine-tuning with rewarded moment matching distillation. *arXiv preprint arXiv:2606.30414*, 2026. URL https://arxiv.org/abs/2606.30414.

Dengyang Jiang, Xin Jin, Dongyang Liu, Zanyi Wang, Mingzhe Zheng, Ruoyi Du, Xiangpeng Yang, Qilong Wu, Zhen Li, Peng Gao, Harry Yang, and Steven Hoi. D-OPSD: On-policy self-distillation for continuously tuning step-distilled diffusion models. *arXiv preprint arXiv:2605.05204*, 2026. URL https://arxiv.org/abs/2605.05204.

Junjie Ke, Qifei Wang, Yilin Wang, Peyman Milanfar, and Feng Yang. MUSIQ: Multi-scale image quality transformer. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 5148–5157, 2021. URL https://openaccess.thecvf.com/content/ICCV2021/html/Ke_MUSIQ_Multi-Scale_Image_Quality_Transformer_ICCV_2021_paper.html.

Gihoon Kim, Hyungjin Park, and Taesup Kim. Preserve and Personalize: Personalized text-to-image diffusion models without distributional drift. In *International Conference on Learning Representations (ICLR)*, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/hash/1a281c4e60aa2184525b4c098ff5bed5-Abstract-Conference.html.

Yuval Kirstain, Adam Polyak, Uriel Singer, Shahbuland Matiana, Joe Penna, and Omer Levy. Pick-a-Pic: An open dataset of user preferences for text-to-image generation. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 36, 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/73aacd8b3b05b4b503d58310b523553c-Abstract-Conference.html.

Yuming Li, Yikai Wang, Yuying Zhu, Zhongyu Zhao, Ming Lu, Qi She, and Shanghang Zhang. BranchGRPO: Stable and efficient GRPO with structured branching in diffusion models. In *International Conference on Learning Representations (ICLR)*, 2026. URL https://openreview.net/forum?id=T2nP2IQasd.

Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matthew Le. Flow matching for generative modeling. In *International Conference on Learning Representations (ICLR)*, 2023. URL https://openreview.net/forum?id=PqvMRDCJT9t.

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. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 38, 2025. URL https://openreview.net/forum?id=oCBKGw5HNf.

Xingchao Liu, Chengyue Gong, and Qiang Liu. Flow straight and fast: Learning to generate and transfer data with rectified flow. In *International Conference on Learning Representations (ICLR)*, 2023. URL https://openreview.net/forum?id=XVjTT1nw5z.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations (ICLR)*, 2019. URL https://openreview.net/forum?id=Bkg6RiCqY7.

Simian Luo, Yiqin Tan, Longbo Huang, Jian Li, and Hang Zhao. Latent consistency models: Synthesizing high-resolution images with few-step inference. *arXiv preprint arXiv:2310.04378*, 2023. URL https://arxiv.org/abs/2310.04378.

Yihong Luo, Tianyang Hu, and Jing Tang. Reinforcing diffusion models by direct group preference optimization. In *International Conference on Learning Representations (ICLR)*, 2026. URL https://openreview.net/forum?id=jymuXl8GYi.

Nanye Ma, Shangyuan Tong, Haolin Jia, Hexiang Hu, Yu-Chuan Su, Mingda Zhang, Xuan Yang, Yandong Li, Tommi Jaakkola, Xuhui Jia, and Saining Xie. Scaling inference time compute for diffusion models. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2523–2534, 2025a. URL https://openaccess.thecvf.com/content/CVPR2025/html/Ma_Scaling_Inference_Time_Compute_for_Diffusion_Models_CVPR_2025_paper.html.

Yuhang Ma, Xiaoshi Wu, Keqiang Sun, and Hongsheng Li. HPSv3: Towards wide-spectrum human preference score. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 15086–15095, 2025b. URL https://openaccess.thecvf.com/content/ICCV2025/html/Ma_HPSv3_Towards_Wide-Spectrum_Human_Preference_Score_ICCV_2025_paper.html.

Zichen Miao, Zhengyuan Yang, Kevin Lin, Ze Wang, Zicheng Liu, Lijuan Wang, and Qiang Qiu. Tuning timestep-distilled diffusion model using pairwise sample optimization. In *International Conference on Learning Representations (ICLR)*, 2025. URL https://openreview.net/forum?id=fXnE4gB64o.

Chong Mou, Yanze Wu, Wenxu Wu, Zinan Guo, Pengze Zhang, Yufeng Cheng, Yiming Luo, Fei Ding, Shiwen Zhang, Xinghui Li, Mengtian Li, Mingcong Liu, Yunsheng Jiang, Shaojin Wu, Songtao Zhao, Jian Zhang, Qian He, and Xinglong Wu. DreamO: A unified framework for image customization. In *ACM SIGGRAPH Asia Conference Papers (SIGGRAPH Asia)*, pp. 1–12, 2025. doi: 10.1145/3757377.3763956. URL https://doi.org/10.1145/3757377.3763956.

Naila Murray, Luca Marchesotti, and Florent Perronnin. AVA: A large-scale database for aesthetic visual analysis. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2408–2415, 2012. doi: 10.1109/CVPR.2012.6247954. URL https://doi.org/10.1109/CVPR.2012.6247954.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning (ICML)*, volume 139 of *Proceedings of Machine Learning Research*, pp. 8748–8763. PMLR, 2021. URL https://proceedings.mlr.press/v139/radford21a.html.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 10684–10695, 2022. URL https://openaccess.thecvf.com/content/CVPR2022/html/Rombach_High-Resolution_Image_Synthesis_With_Latent_Diffusion_Models_CVPR_2022_paper.html.

Aniket Roy, Shubhankar Borse, Shreya Kadambi, Debasmit Das, Shweta Mahajan, Risheek Garrepalli, Hyojin Park, Ankita Nayak, Rama Chellappa, Munawar Hayat, and Fatih Porikli. DuoLoRA: Cycle-consistent and rank-disentangled content-style personalization. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 15395–15404, 2025. URL https://openaccess.thecvf.com/content/ICCV2025/html/Roy_DuoLoRA__Cycle-consistent_and_Rank-disentangled_Content-Style_Personalization_ICCV_2025_paper.html.

Nataniel Ruiz, Yuanzhen Li, Varun Jampani, Yael Pritch, Michael Rubinstein, and Kfir Aberman. DreamBooth: Fine tuning text-to-image diffusion models for subject-driven generation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 22500–22510, 2023. URL https://openaccess.thecvf.com/content/CVPR2023/html/Ruiz_DreamBooth_Fine_Tuning_Text-to-Image_Diffusion_Models_for_Subject-Driven_Generation_CVPR_2023_paper.html.

Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, Patrick Schramowski, Srivatsa Kundurthy, Katherine Crowson, Ludwig Schmidt, Robert Kaczmarczyk, and Jenia Jitsev. LAION-5B: An open large-scale dataset for training next generation image-text models. In *Advances in Neural Information Processing Systems (NeurIPS), Datasets and Benchmarks Track*, volume 35, 2022. URL https://proceedings.neurips.cc/paper_files/paper/2022/hash/a1859debfb3b59d094f3504d5ebb6c25-Abstract-Datasets_and_Benchmarks.html.

Idan Shenfeld, Mehul Damani, Jonas Hübotter, and Pulkit Agrawal. Self-distillation enables continual learning. In *International Conference on Machine Learning (ICML)*, 2026. URL https://openreview.net/forum?id=qA6FgH0nnZ.

Oriane Siméoni, Huy V. Vo, Maximilian Seitzer, Federico Baldassarre, Maxime Oquab, Cijo Jose, Vasil Khalidov, Marc Szafraniec, Seungeun Yi, Michaël Ramamonjisoa, Francisco Massa, Daniel Haziza, Luca Wehrstedt, Jianyuan Wang, Timothée Darcet, Théo Moutakanni, Leonel Sentana, Claire Roberts, Andrea Vedaldi, Jamie Tolan, John Brandt, Camille Couprie, Julien Mairal, Hervé Jégou, Patrick Labatut, and Piotr Bojanowski. DINOv3. *Transactions on Machine Learning Research (TMLR)*, 2026. URL https://openreview.net/forum?id=2NlGyqNjns.

Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In *International Conference on Learning Representations (ICLR)*, 2015. URL https://arxiv.org/abs/1409.1556.

Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. In *International Conference on Learning Representations (ICLR)*, 2021a. URL https://openreview.net/forum?id=St1giarCHLP.

Yang Song, Jascha Sohl-Dickstein, Diederik P. Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. In *International Conference on Learning Representations (ICLR)*, 2021b. URL https://openreview.net/forum?id=PxTIG12RRHS.

Yang Song, Prafulla Dhariwal, Mark Chen, and Ilya Sutskever. Consistency models. In *International Conference on Machine Learning (ICML)*, pp. 32211–32252, 2023. URL https://proceedings.mlr.press/v202/song23a.html.

Ruoyu Wang, Boye Niu, Xiangxin Zhou, Yushi Huang, Tongliang Liu, and Chi Zhang. Exploring the design space of reward backpropagation for flow matching. *arXiv preprint arXiv:2606.11075*, 2026. URL https://arxiv.org/abs/2606.11075.

Xiaoshi Wu, Yiming Hao, Keqiang Sun, Yixiong Chen, Feng Zhu, Rui Zhao, and Hongsheng Li. Human preference score v2: A solid benchmark for evaluating human preferences of text-to-image synthesis. *arXiv preprint arXiv:2306.09341*, 2023. URL https://arxiv.org/abs/2306.09341.

Jiazheng Xu, Xiao Liu, Yuchen Wu, Yuxuan Tong, Qinkai Li, Ming Ding, Jie Tang, and Yuxiao Dong. ImageReward: Learning and evaluating human preferences for text-to-image generation. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 36, 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/33646ef0ed554145eab65f6250fab0c9-Abstract-Conference.html.

Zeyue Xue, Jie Wu, Yu Gao, Fangyuan Kong, Lingting Zhu, Mengzhao Chen, Zhiheng Liu, Wei Liu, Qiushan Guo, Weilin Huang, and Ping Luo. DanceGRPO: Unleashing GRPO on visual generation. *arXiv preprint arXiv:2505.07818*, 2025. URL https://arxiv.org/abs/2505.07818.

Yixiong Yang, Tao Wu, Senmao Li, Shiqi Yang, Yaxing Wang, Joost van de Weijer, and Kai Wang. Adversarial concept distillation for one-step diffusion personalization. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition Findings (CVPR Findings)*, pp. 4321–4333, 2026. URL https://openaccess.thecvf.com/content/CVPR2026F/html/Yang_Adversarial_Concept_Distillation_for_One-Step_Diffusion_Personalization_CVPRF_2026_paper.html.

Tianwei Yin, Michaël Gharbi, Taesung Park, Richard Zhang, Eli Shechtman, Frédo Durand, and William T. Freeman. Improved distribution matching distillation for fast image synthesis. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 37, 2024a. URL https://papers.nips.cc/paper_files/paper/2024/hash/54dcf25318f9de5a7a01f0a4125c541e-Abstract-Conference.html.

Tianwei Yin, Michaël Gharbi, Richard Zhang, Eli Shechtman, Frédo Durand, William T. Freeman, and Taesung Park. One-step diffusion with distribution matching distillation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 6613–6623, 2024b. URL https://openaccess.thecvf.com/content/CVPR2024/html/Yin_One-step_Diffusion_with_Distribution_Matching_Distillation_CVPR_2024_paper.html.

Zhiyuan You, Xin Cai, Jinjin Gu, Tianfan Xue, and Chao Dong. Teaching large language models to regress accurate image quality scores using score distribution. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025. URL https://arxiv.org/abs/2501.11561.

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. URL https://arxiv.org/abs/2511.22699.

Richard Zhang, Phillip Isola, Alexei A. Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 586–595, 2018. URL https://openaccess.thecvf.com/content_cvpr_2018/html/Zhang_The_Unreasonable_Effectiveness_CVPR_2018_paper.html.

Yihua Zhang, Chongyu Fan, Yimeng Zhang, Yuguang Yao, Jinghan Jia, Jiancheng Liu, Gaoyuan Zhang, Gaowen Liu, Ramana Kompella, Xiaoming Liu, and Sijia Liu. UnlearnCanvas: Stylized image dataset for enhanced machine unlearning evaluation in diffusion models. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 37, pp. 96387–96423, 2024. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/aebf4822d30c3f2600566af7eba83548-Abstract-Datasets_and_Benchmarks_Track.html.

Siyan Zhao, Zhihui Xie, Mengchen Liu, Jing Huang, Guan Pang, Feiyu Chen, and Aditya Grover. Self-distilled reasoner: On-policy self-distillation for large language models. In *International Conference on Machine Learning (ICML)*, 2026. URL https://openreview.net/forum?id=Jpxfof0EaS.

Kaiwen Zheng, Huayu Chen, Haotian Ye, Haoxiang Wang, Qinsheng Zhang, Kai Jiang, Hang Su, Stefano Ermon, Jun Zhu, and Ming-Yu Liu. DiffusionNFT: Online diffusion reinforcement with forward process. In *International Conference on Learning Representations (ICLR)*, 2026. URL https://openreview.net/forum?id=VJZ477R89F.

Wei Zhou, Xiongwei Zhu, Lingdong Kong, Bo Chen, Lei Zhang, Yongyuan Liang, Xiaoxia Hou, Ye Tian, Xian Sun, Yingshuo Wang, Linfeng Li, Shengqiong Wu, Leigang Qu, Feng Li, Wei Liu, Julian McAuley, and Tat-Seng Chua. On-policy self-distillation in diffusion models. *arXiv preprint arXiv:2608.24646*, 2026. URL https://arxiv.org/abs/2608.24646.

Table of contents:

- §A: Notation
- §B: Gradient Interpretation and State Weighting
- §C: Reward Definition and Evaluation Scope
- §D: Schedules, Hyperparameters, and Computation
- §E: Experimental Protocol
- §F: Reference-Free Preference and Quality Scorers
- §G: Supplementary Human Preference Study
- §H: Training and Validation Curves
- §I: Limitations
- §J: Additional Qualitative Comparisons

A NOTATION

Time t decreases along sampling from $t_K = 1$ (noise) to $t_0 = 0$ (data). The reference image x_0 belongs to a training pair (x_0, y), and the final latent is $x^s_{t_0}$. States are indexed $k = K, \dots, 1$ in the order they are visited, so the final visited state is $k = 1$.

Table S1: **Principal symbols for the method and its analysis.** Continuations use the student fixed during target construction.

Symbol	Meaning
x_0, y	Reference image and prompt of a pair drawn from training distribution $\mathcal{D}$
c_s, c_T	Student (text) and teacher (multimodal) conditions
$x^s_{t_k}, t_k$	Visited student state and its time
u^s_k, u^T_k	Student and EMA-teacher velocities at the visited state
$\hat{x}^s_k, \hat{x}^T_k$	Student prediction and teacher target, $x^s_{t_k} - t_k u_k$
$x^s_{t_0}$	Final latent, $x^s_{t_0} = \hat{x}^s_1$
$C_k(\hat{x})$	Continuation: the final latent after replacing the state-k prediction by $\hat{x}$
B_k, g_k	Continuation Jacobian at $\hat{x}^s_k$ and the reward gradient $\nabla R(\hat{x}^s_k)$ at the prediction
b_k, d_k, x^+_k	Radius $\delta\|\hat{x}^s_k\|_2$, unit direction $g_k/\|g_k\|_2$, and modified prediction $\hat{x}^s_k + b_k d_k$
c_k	Euler coefficient $(t_k - t_{k-1})/t_k$ and SRA reward regression weight
J_k	Direct parameter Jacobian of the prediction at fixed $x^s_{t_k}$
κ, λ	Reward-to-distillation gradient-norm ratio and its balancing coefficient
$\mathcal{L}_D, \mathcal{L}_{\text{aux}}$	Expected distillation and reward-regression losses, with hats denoting minibatch estimates
$F_k(\hat{x})$	Reward $R(C_k(\hat{x}))$ under the fixed student's continuation

B GRADIENT INTERPRETATION AND STATE WEIGHTING

We characterize the joint update through three choices: local reward-gradient rescaling, relative state weights, and aggregate gradient-norm balancing. The identities below describe how local reward supervision contributes when combined with reference-conditioned teacher supervision.

B.1 LOCAL AND CONTINUATION REWARD GRADIENTS

Fix a training pair, initial noise, schedule, and student parameters at target construction. Let $C_k(s, \hat{x})$ denote the final latent after using $\hat{x}$ in the Euler transition at state s and step k, then recomputing the remaining trajectory with this fixed student. Define $F_k(s, \hat{x}) = R(C_k(s, \hat{x}))$ and omit the fixed state and reward conditions below.

Assume replay reproduces the construction predictions. With J_k denoting the prediction's direct parameter Jacobian at fixed input state, the chain rule gives the full reward gradient:

$$\nabla_\theta R(x^s_{t_0}) = \sum_{k=1}^{K} J_k^\top \nabla F_k(\hat{x}^s_k). \tag{S1}$$

Each continuation derivative includes downstream state responses. Let ℓ_{aux} denote the auxiliary loss for this fixed pair and noise draw, whose expectation is $\mathcal{L}_{\text{aux}}$. For stopped targets and λ, fitting gives

$$-\nabla_\theta \ell_{\text{aux}} = \frac{1}{K} \sum_{k:\, g_k \neq 0} c_k b_k J_k^\top d_k = \frac{1}{K} \sum_{k:\, g_k \neq 0} \frac{c_k b_k}{\|g_k\|_2} J_k^\top g_k, \tag{S2}$$

where $g_k = \nabla R(\hat{x}_k^s)$ is the local reward gradient. Target fitting combines normalized local gradients with prediction-norm scaling and Euler weights. At fixed parameters and detached rollout states, using the same local reward gradients g_k, the Euler-weighted ReFL ascent direction is proportional to $\sum_k c_k J_k^\top g_k$. SRA additionally scales each nonzero local contribution by $\|\hat{x}_k^s\|_2 / \|g_k\|_2$ before aggregate norm balancing. This local rescaling and the Euler weights determine the relative contributions. Under exact norm balancing, κ sets their aggregate norm relative to the distillation gradient. Local and continuation derivatives coincide at the final prediction, where $C_1(\hat{x}) = \hat{x}$, and can differ earlier. Matching total gradients, parameters, and optimizer state give identical updates without guaranteeing reward improvement.

The implementation averages each squared target error over its N latent entries. If h_k^A is the parameter gradient of the Euler-weighted ReFL surrogate, including c_k/K, the corresponding target-fit gradient on the same cached reward gradient satisfies $h_k^C = (a_k/N) h_k^A$ in exact arithmetic, with $a_k = \delta \|\hat{x}_k^s\|_2 / \|g_k\|_2$ for $g_k \neq 0$. The common $1/N$ accounts for the MSE mean and cancels from the balanced reward contribution under the conditions below. Appendix E.4 checks this relation numerically with shared reward gradients.

B.2 Euler weights and common scales

The immediate Euler transition contributes $c_k I$. If S_k maps the next state to the final latent, the continuation Jacobian is

$$B_k = \frac{\partial S_k}{\partial x^s_{t_{k-1}}} c_k I. \tag{S3}$$

Later predictions can rotate, contract, or amplify a perturbation. Thus c_k alone neither ranks final reward changes nor bounds the norm of B_k by one.

With one update per target construction, a common positive scaling of δ or all reward regression weights scales the auxiliary gradient and inversely scales λ, leaving their product unchanged. This assumes fixed states, parameters, and directions, matching replay predictions, nonzero gradient norms, and exact arithmetic with numerical floors inactive and stabilizers omitted (Appendices D.3 and D.4). The sampling schedule stays fixed. Relative state weights and prediction norms can still affect direction. Thus δ sets target displacement and κ the balanced raw-gradient norm ratio. Finite precision, repeated fitting, multi-step target construction, and variants without this balancing rule need not preserve the cancellation.

C Reward Definition and Evaluation Scope

Let $\mathcal{V}$ be the image decoder. On Z-Image-Turbo, SRA uses the following reward.

$$R(x, x_0, y) = 21.6\, R_{\text{DINOv3}}(\mathcal{V}(x), x_0) + 12.8\, R_{\text{TOPIQ}}(\mathcal{V}(x)). \tag{S4}$$

The first component is DINOv3 cosine similarity to the reference image (Siméoni et al., 2026). The second is TOPIQ no-reference image quality (Chen et al., 2024). Neither explicitly scores prompt adherence. The prompt enters the student and the privileged teacher, while adherence is assessed separately.

Table S2: **Reward components and fixed scales on Z-Image-Turbo.**

Component	Model	Scale
DINOv3 (Siméoni et al., 2026)	`dinov3_vits16plus`	21.6
TOPIQ (Chen et al., 2024)	`topiq_nr`	12.8

Table S2 reports the Z-Image-Turbo scales established by the latent-gradient calibration. For FLUX.2-klein-4B, the DINOv3 and TOPIQ coefficients are approximately 118.7526 and 49.5871, respectively.

They are recalibrated as the reciprocals of the median component gradient norms in the model's latent space over 100 panels at initialization. These two components were selected during development and need not suffice for every personalization objective. Distillation does not guarantee prompt adherence.

The scorers and decoder are fixed during differentiation. Component image gradients are accumulated and propagated through the decoder to form the latent gradient. SRA evaluates the latent reward gradient at every visited prediction.

Shared scorer families. DINO-D, DINOv3, and Quality-S overlap with the DINOv3 or TOPIQ training families and are reward-related diagnostics for our method. LPIPS-D, VLM-J, CLIP-S, and Aesthetic-S are outside those two families, subject to the qualifications in Appendix E. A scorer outside the training families is not automatically statistically independent or universally valid as a measure of human preference.

C.1 Reward recipes

Table 5 trains uniform-weight ReFL, DRaFT, and SRA on top of D-OPSD under four further rewards and reports, for each recipe, only the metrics that recipe rewards. The second recipe replaces TOPIQ in Equation S4 by PickScore (Kirstain et al., 2023) at scale 0.4785, calibrated by the same latent-gradient rule as the other scales (Table S2). The remaining three reward a single term: DINOv3 alone at scale 21.6, TOPIQ alone at 12.8, and ImageReward (Xu et al., 2023) alone at 0.9275, the last calibrated by the same rule on 100 panels at initialization. Because the reward gradient is balanced against the distillation gradient by norm, a single term's scale changes only the logged reward values, not the update. Every other setting is that of Table 1: one student chain per optimizer step, 1,000 updates, the same reward-to-distillation gradient-norm ratio, and the same twenty cases and seed realization. The D-OPSD row is the same set of checkpoints in every group because D-OPSD uses no reward. PickScore and ImageReward are read with the scorers of Appendix F, which are the rewarded models.

Within every recipe read so far, SRA reaches the largest value of each rewarded metric, and its paired differences against the two baselines exclude zero on the identity metrics and on the reward's own target: under the PickScore recipe SRA $-$ DRaFT is $+0.034$ on DINO-D, $+0.028$ on DINOv3, and $+0.48$ on PickScore (SRA $-$ ReFL: $+0.029$, $+0.014$, and $+0.43$). With DINOv3 the only rewarded term, SRA $-$ DRaFT is $+0.038$ on DINO-D and $+0.042$ on DINOv3 (SRA $-$ ReFL: $+0.031$ and $+0.031$), the same margins as under the main recipe.

D Schedules, Hyperparameters, and Computation

D.1 Training and deployment schedules

Table S3 lists the Z-Image-Turbo schedules. FLUX.2-klein-4B uses its native four-step sampler for both training and deployment, with visited times 1, 0.967384, 0.908144, 0.767200 and a final transition to $t = 0$.

Table S3: **Sampling schedules.** Listed are the times $t_K, \dots, t_1$ of the visited states, with $t_K = 1$ denoting noise. The last transition ends at $t_0 = 0$. Deployment values are rounded for display.

Use	Visited times t	Effective transitions
Training	1, 0.9, 0.75, 0.5	4
Deployment	1, .95455, .9, .83333, .75, .64286, .5, .3	8

Z-Image-Turbo deployment additionally evaluates the model at $t = 0$ with a zero-length transition, giving nine actual model forwards. The description *nine-step sampler* counts this call, whereas the eight effective transitions count only latent changes. The teacher panel uses the same deployment convention. Training returns its last prediction $\hat{x}_1^s$, the final latent after the fourth effective transition.

For Z-Image-Turbo, the training schedule uses a subset of the deployment times, but states reached at a common time generally differ between schedules. Our method constructs its targets on the four-transition training chain. Performance under deployment is a separate empirical question.

D.2 Training procedure

Algorithm 1 lists one training update of SRA for a single training example. Before training, the student and EMA teacher are initialized from the same pretrained weights, and the Euler weights c_k are computed from the fixed schedule using Eq. (4). Each update samples a training pair and fresh initial noise. The loss estimates $\widehat{\mathcal{L}}_D$ and $\widehat{\mathcal{L}}_{\text{aux}}$ in the algorithm use this pair and noise draw in place of the expectations in Eqs. (2) and (5). For larger minibatches, both losses are averaged over examples before the gradient-norm ratio is formed.

Sampling transitions retain no parameter graph: the chain advances from the detached student prediction. Each detached clean prediction is treated as a separate input to the decoder and reward model, with differentiation enabled only with respect to that input, in single precision, with the safeguards of Appendix D.4. Parameter gradients flow only through the student predictions at the visited states. Both losses are aggregated over states before forming λ and applying one student update, followed by one EMA update. No gradient crosses a sampling step. Appendix D.3 specifies the numerical safeguards for λ.

Parameters and teacher. The trained parameters θ are the LoRA adapters of Table S4. The base weights are frozen. The teacher is the EMA of the same adapters on the same base, conditioned additionally on the reference image, and is initialized at the student's initialization. With decay $\mu = 0.9999$ over 1,000 updates, the initialization retains a weight of $0.9999^{1000} \approx 0.90$ in the teacher, so the teacher stays close to the reference-conditioned base model throughout training.

Two-pass implementation. The implementation runs the chain twice from the same initial noise: a first pass without any parameter graph that constructs x_k^+, and a second pass with the student forward carrying the graph and the state detached between steps, which recomputes the same predictions under deterministic kernels and fits both losses. It records at every update the relative root-mean-square difference between the two passes' predictions ($< 10^{-3}$ required, 0 observed on every cell). With matching predictions and the same local reward gradients, the single pass of Algorithm 1, which builds each target from the detached prediction of the graph pass, is algebraically equivalent to the two-pass implementation, up to floating-point effects, at four fewer student forwards per update.

Recorded endpoint check. The shipped configuration also computes, for each local reward update, whether the move improves the reward at the end of the chain: after replacing $\hat{x}_k^s$ by x_k^+ it continues the remaining sampling steps without gradient and scores the endpoint. The outcome is recorded and consumed by no weight. The check costs six additional student forwards per update (the continuations from states $K, \dots, 2$).

Computational work and measured time. Each update evaluates one local reward gradient at each of the four states; checkpointing may add recomputation. On the H200 runs of Appendix H, including the diagnostic continuations above, SRA takes 2.12 GPU hours per 1,000 updates. Uniform-weight ReFL takes the same four reward gradients on the same chain, and its measured 2.05 hours is within 0.1 hours.

D.3 Optimization

Table S4 gives the Z-Image-Turbo settings. FLUX.2-klein-4B uses a learning rate of 10^{-5} with the same remaining settings. Both models use LoRA (Hu et al., 2022) adaptation of attention and feed-forward projections. The squared norms in Eqs. (2) and (5) are implemented as means over latent coordinates. The auxiliary term includes the $1/2$, and both terms are averaged over the four states. We denote the minibatch estimates of the two losses by $\widehat{\mathcal{L}}_D$ and $\widehat{\mathcal{L}}_{\text{aux}}$. The balancing coefficient is computed from aggregated raw parameter gradients before clipping and AdamW (Loshchilov & Hutter, 2019) preconditioning.

Table S4: **Protocol parameters for SRA.** All configurations in Table 2 use one student chain per optimizer step.

Parameter	Value
Relative target radius δ	0.07
Reward-to-distillation gradient-norm ratio κ	0.3
Student chains per optimizer step / states per chain	1 / 4
Optimizer steps	1,000
Optimizer / learning rate	AdamW / 4×10^{-5}
AdamW weight decay / epsilon	0 / 10^{-8}
LoRA rank / alpha / dropout	64 / 128 / 0
Gradient clipping norm	1.0
EMA decay	0.9999

At each update, λ is zero if the auxiliary gradient norm is below 10^{-6} times the distillation gradient norm. It is also zero if either norm is at most 10^{-12}. Otherwise, we set

$$\lambda = \kappa \frac{\|\nabla_\theta \widehat{\mathcal{L}}_D\|_2}{\|\nabla_\theta \widehat{\mathcal{L}}_{\text{aux}}\|_2}.$$

The coefficient is held fixed when combining the two gradients for the student update. These floors prevent cancellation noise from producing a large balancing coefficient.

Choice of κ. The ratio $\kappa = 0.3$ was fixed when the gradient balancing was introduced and was not changed afterwards. Appendix E.4 reports the strength sweeps on both base models.

D.4 Numerical implementation

Gradient norms accumulate in float64 and are converted to the gradient dtype before adding a 10^{-12} stabilizer. Nonfinite gradient entries are replaced by zero before normalization and counted. A zero gradient leaves the prediction unchanged. Floating-point reassociation and dtype conversion can affect prediction replay (Appendix D.2).

The reward-gradient repeatability diagnostic in Appendix E.4 additionally tests `torch.use_deterministic_algorithms(True, warn_only=True)` and `torch.backends.cudnn.deterministic=True`; the former allows an operation without a deterministic implementation to proceed with a warning. Gradient sharing is used only for this diagnostic; training retains its original per-state reward-gradient calls.

E Experimental Protocol

The validation set comprises twenty personalization cases, ten objects from the DreamBooth dataset (Ruiz et al., 2023) and ten styles from UnlearnCanvas (Zhang et al., 2024). Each case has a train set of four reference and prompt pairs. Each pair contains one image, and each teacher query uses one such reference–prompt pair from the case's train set. The case suite has informed method development, and its results are descriptive validation evidence. A separate test set contains eight new cases evaluated after fixing the method and hyperparameters (Appendix E.2).

We evaluate the fixed final checkpoint after 1,000 updates. No separate split is used for checkpoint selection. For each case and method, the evaluation panel contains twenty generated images: one rewrite per reference row (four images), and four evaluation prompts per row (sixteen images). This produces 400 images across the twenty cases for one method, with paired prompt, reference, and noise coordinates.

The prompts come from two sources, both fixed before training and pinned by digest. For the ten object cases they are DreamBooth templates: the rewrite prompt is the training sentence itself, and the four evaluation prompts are a fixed selection from that template list, so no model writes them. The templates carry the DreamBooth placeholder `[V]`. We render it to the class noun for every method, so the training sentence reads "photo of a dog" and an evaluation prompt "a dog in the jungle". No method trains an identifier token: the self-distillation family carries subject identity through the

reference image and the adapter weights, and we keep one prompt set for all compared methods. The same rendering applies to the style templates. For the ten style cases, which the source dataset ships without prompts, both roles were written once by Qwen3-VL-8B-Instruct (Bai et al., 2025) through the prompt generator and output validator of the D-OPSD reference implementation, with greedy decoding and a declared temperature and seed fallback for rows greedy decoding could not make valid. The same model family provides VLM-J, so the judge and prompt author are the same model on style cases. VLM-J compares an image with its reference without conditioning on the prompt. The prompts are identical across the compared methods. This overlap remains a limitation of evaluation on style cases.

Metric definitions. Table S5 defines the seven reported metrics. All scorers are frozen during evaluation.

Table S5: **Evaluation metrics and scorer models.**

Metric	Definition	Scorer and source
DINO-D ↓	One minus cosine similarity to the corresponding reference.	DINOv3 ViT-S/16+ (Siméoni et al., 2026)
LPIPS-D ↓	Perceptual distance to the corresponding reference.	LPIPS (Zhang et al., 2018) with VGG features (Simonyan & Zisserman, 2015)
VLM-J ↑	Reference-based object identity or style consistency, rated from 1 to 4.	Qwen3-VL-8B-Instruct (Bai et al., 2025)
CLIP-S ↑	Cosine similarity between image and evaluation-prompt embeddings.	DFN5B-CLIP-ViT-H-14 (Radford et al., 2021, Fang et al., 2024)
DINOv3 ↑	Maximum cosine similarity across the four reference images.	DINOv3 ViT-S/16+ (Siméoni et al., 2026)
Quality-S ↑	No-reference image quality score.	TOPIQ-NR (Chen et al., 2024), KonIQ checkpoint (Hosu et al., 2020)
Aesthetic-S ↑	Mean aesthetic rating from 1 to 10.	MUSIQ (Ke et al., 2021), AVA checkpoint (Murray et al., 2012)

Both DINO metrics use the same ViT-S/16+ LVD-1689M weights and average patch tokens before ℓ_2 normalization. DINO-D and LPIPS-D resize the reference to the generated image size. DINOv3 resizes both images to 512×512. LPIPS-D uses the v0.1 VGG weights with images scaled to $[-1, 1]$. CLIP-S uses the unscaled cosine similarity. Quality-S and Aesthetic-S use the `topiq_nr` and `musiq-ava` models in `pyiqa`, respectively.

VLM-J compares each image with its paired reference using deterministic decoding and separate object and style rubrics. The object labels are *different instance*, *partly similar*, *mostly same*, and *same instance*. The style labels are *not consistent*, *slightly consistent*, *mostly consistent*, and *highly consistent*. Each list maps to scores from 1 to 4 in the given order. The style rubric permits changes in subject and composition.

Table S6: **Images used by the terminal instruments within each case.** Different instruments do not all average the same twenty images.

Measurements	Images per case
DINO-D, LPIPS-D	4 rewrites
CLIP-S	All 20 images
VLM-J, DINOv3, Quality-S, Aesthetic-S	16 evaluation-prompt images

Cases, rather than images or intermediate states, are the units of final performance comparisons. Object and style means are reported separately. Mechanism-probe counts describe fixed-model interventions and do not add independent training replicates.

Evaluation constructs. DINO-D and LPIPS-D are lower-is-better correspondence distances. The other five metrics are higher-is-better scores. Object-mode VLM judgments pass the study's construct qualification, whereas the style cross-reference construct does not. A rise in style VLM-J cannot on its own be called improved style preservation. Reward-family overlap is distinct from these qualifications (Appendix C).

E.1 TRAINING-SEED REPLICATION

Every result in Tables 1 and 2 comes from one training run per method and case, at the protocol's seed realization r_0. To measure how much of a paired difference is training-seed variation, SRA, D-OPSD, DRaFT, and uniform-weight ReFL were trained twice more on all twenty cases of each base model at two further realizations r_1 and r_2 of the seed formula, which redraws every seeded quantity (initialization, training rows, prompts, noises, and generation seeds) while keeping the schedule, optimizer, and reward fixed. Table 6 includes all four methods' three-seed means and sample standard deviations on both models. Table S7 reports, for the three reward-related metrics, the smallest paired effect over the three realizations and the effect pooled over seeds. The pooled estimate first averages each case's effect over its three realizations, then bootstraps cases within object and style groups with equal group weights, using 20,000 draws. These pointwise intervals describe variation across cases after seed averaging; they are not intervals over a population of training seeds and are not adjusted for multiple comparisons.

All three reward-related differences between SRA and each of D-OPSD, uniform-weight ReFL, and DRaFT favor SRA at every seed on both models, and all eighteen pooled intervals exclude zero. Among the other metrics, the consistent cost is CLIP-S. On FLUX.2-klein-4B it is lower than all three baselines at every seed, with pooled differences of -0.0045, -0.0017, and -0.0035 against D-OPSD, ReFL, and DRaFT, each interval below zero. On Z-Image-Turbo only the difference from D-OPSD has an interval below zero (-0.0039). The Euler-weighted ReFL and DRaFT ablations are separate single-seed comparisons and are not included in these seed summaries.

Table S7: **Paired differences pooled over three training seeds.** Differences are the first minus the second method in each metric's native direction, and arrows give the better direction. The pooled estimate averages each case over its three seeds before the stratified case bootstrap, and brackets are pointwise 95% intervals. ReFL and DRaFT use uniform weights. All 54 single-seed differences between SRA and a baseline favor SRA, the smallest being 0.0095 in DINOv3 against ReFL on FLUX.2-klein-4B.

Pair	DINO-D↓	DINOv3↑	Quality-S↑
Z-Image-Turbo			
SRA − D-OPSD	−0.0777 [−0.0887, −0.0663]	+0.0500 [+0.0406, +0.0594]	+0.1937 [+0.1787, +0.2084]
SRA − ReFL (uniform)	−0.0318 [−0.0377, −0.0256]	+0.0244 [+0.0180, +0.0310]	+0.0880 [+0.0796, +0.0965]
SRA − DRaFT	−0.0304 [−0.0385, −0.0229]	+0.0220 [+0.0170, +0.0279]	+0.0804 [+0.0665, +0.0946]
DRaFT − D-OPSD	−0.0472 [−0.0601, −0.0351]	+0.0280 [+0.0203, +0.0360]	+0.1133 [+0.0985, +0.1280]
FLUX.2-klein-4B			
SRA − D-OPSD	−0.0557 [−0.0659, −0.0452]	+0.0262 [+0.0210, +0.0323]	+0.2500 [+0.2200, +0.2798]
SRA − ReFL (uniform)	−0.0243 [−0.0290, −0.0198]	+0.0109 [+0.0079, +0.0145]	+0.0560 [+0.0486, +0.0638]
SRA − DRaFT	−0.0249 [−0.0299, −0.0198]	+0.0135 [+0.0101, +0.0174]	+0.0709 [+0.0611, +0.0814]
DRaFT − D-OPSD	−0.0308 [−0.0378, −0.0239]	+0.0127 [+0.0089, +0.0167]	+0.1792 [+0.1551, +0.2031]

E.2 NEW TEST CASES

The test set comprises four further DreamBooth subjects (dog, duck toy, fancy boot, and grey sloth plushie) and four further UnlearnCanvas styles (Crayon, Crypto Punks, Cubism, and Dadaism). The objects are the next four staged subjects in dictionary order with distinct class names. The styles are the next four eligible styles after the validation roster. The method, reward, hyperparameters, and 1,000-update policy were fixed before evaluating these cases. Each case has a four-reference train set and the same twenty-image evaluation panel and metric-specific subsets as the validation cases. No additional validation set or checkpoint selection is used within a test case.

The four methods, including uniform-weight ReFL, are adapted separately to every test case on Z-Image-Turbo, with matched prompt, reference, and noise coordinates and one training seed per method and case. Table S8 reports the paired differences on the three reward-related metrics. We average within each case, give equal weight to the object and style means, and use 20,000 bootstrap draws resampling the four cases within each group. All nine favor SRA with intervals excluding zero. Among the other metrics, LPIPS-D is higher than for D-OPSD and ReFL by 0.0074 [0.0038, 0.0106] and 0.0059 [0.0018, 0.0101], all CLIP-S and VLM-J intervals include zero, and Aesthetic-S improves over D-OPSD by 0.37 [0.19, 0.60].

Table S8: **Paired differences on the eight test cases.** Differences are SRA minus each baseline in each metric's native direction, and arrows give the better direction. Brackets give pointwise 95% case-bootstrap intervals. Each method has one training run per case.

Comparison	DINO-D↓	DINOv3↑	Quality-S↑
SRA − D-OPSD	−0.0636 [−0.0829, −0.0472]	+0.0531 [+0.0450, +0.0623]	+0.1714 [+0.1550, +0.1870]
SRA − ReFL (uniform)	−0.0342 [−0.0492, −0.0223]	+0.0229 [+0.0174, +0.0267]	+0.0856 [+0.0759, +0.0945]
SRA − DRaFT	−0.0325 [−0.0452, −0.0213]	+0.0234 [+0.0124, +0.0335]	+0.0724 [+0.0514, +0.0925]

E.3 Baselines in the main table

The baselines follow the comparison of D-OPSD (Jiang et al., 2026) and use the D-OPSD training conditions of each base model. *D-OPSD* uses teacher distillation alone. Our *ReFL* adaptation maximizes rewards at every detached student prediction with uniform state weights. Our *DRaFT* adaptation differentiates the final generated-image reward through the full sampling trajectory, with uniform weights on the states' parameter-gradient contributions. Both retain the same distillation objective and aggregate gradient-norm balancing rule. The main table, seed replication, test cases, reward recipes, preference panel, and checkpoint curves use these uniform-weight adaptations; Euler-weighted variants are reported in Appendix E.4. These comparisons evaluate reward-update rules within the shared OPSD protocol.

Base model is the step-distilled model without adaptation. *SFT* regresses the flow-matching velocity at noised reference images (Liu et al., 2023, Lipman et al., 2023). *DreamBooth* (Ruiz et al., 2023) adds a prior-preservation loss on 100 class images per case, generated by the unadapted step-distilled model. We run it with the shared class-noun prompts rather than a trained identifier token, so the subject and the class images share the same prompt. This is weaker than its canonical form. A two-case check of the identifier variant on Z-Image-Turbo, inserting `sks` before the class noun in the subject and sampling prompts, did not improve the baseline: the unadapted model renders the token as text, and after the same 1,000 updates the reference similarity was no higher than with class-noun prompts. PSO inherits the same condition through its prior-preservation term. *PSO* (Miao et al., 2025) is adapted to this setting. It prefers noised reference images over the four states of the student's own chain in a pairwise loss and keeps the prior-preservation term. *SFT→LoRA* trains a LoRA (Hu et al., 2022) with SFT on the multi-step base model, Z-Image (Z-Image Team et al., 2025) or FLUX.2-klein-base-4B (Black Forest Labs, 2025, Black Forest Labs et al., 2025), and loads it into the step-distilled model without further training.

E.4 Configurations and interpretation of the ablations

Table 2 uses twenty Z-Image-Turbo cases, one four-state chain per update, 1,000 updates, and the main composite reward. Unless specified below, teacher supervision and the aggregate reward-to-distillation gradient-norm ratio follow SRA. Target-fitting variants retain its radius rule and four-state loss denominator. Configuration definitions follow the groups in Table 2. Paired training comparisons are followed by fixed-checkpoint diagnostics of reward-term parameter gradients and rescaling consistency.

Reward construction and state weighting. The first four rows in Tables 2 and S9 cross ReFL and one-sided targets with uniform and Euler weights.

- *ReFL, uniform* uses per-state direct reward gradients with unit weights, as in Table 1.
- *ReFL, Euler* applies Euler coefficients to ReFL's per-state direct reward gradients before aggregation and gradient-norm balancing.
- *One-sided, uniform* fits the same local targets as SRA with unit weights. The one-sided weighting variants retain prediction-norm scaling and reward-gradient normalization.
- *One-sided, Euler (SRA)* fits one normalized local reward-ascent target at every state with Euler weights.
- *One-sided, last-heavy* uses the same targets with weights $[0.2, 0.2, 0.2, 1]$ in rollout order on Z-Image-Turbo.

- *Two-sided, uniform* uses half the difference between two-step reward-ascent and reward-descent proposals as its target displacement, with uniform weights and three reward gradients per state (Zhou et al., 2026).
- *Two-sided, Euler* applies Euler weights to the same two-sided construction.
- *DRaFT, Euler* multiplies DRaFT's per-state contributions by Euler coefficients before aggregating and balancing their parameter gradients.

State coverage.

- *Final state only* uses reward-fitting weights $[0, 0, 0, 1]$ in rollout order, retaining teacher supervision at all states.
- *Final-state-only with gain verification* fits the final target only for a positive endpoint gain and a finite direction, in the post hoc comparison of Table S10.

Gain-weighted targets.

- *Backpropagated* fits bounded targets in the normalized terminal-reward pullback direction using the gain weights in Equation S6.
- *Random* replaces the pullback direction with normalized Gaussian noise, using the same gain-weighting rule.
- *Two-sided* in the gain-weighted group fits the two-sided targets using the gain weights.

Shared gain-weighting rule. With the student fixed, endpoint gains relative to the original rollout and the corresponding fitting weights are

$$G_k = R(C_k(x_k^+)) - R(x_{t_0}^s). \tag{S5}$$

$$w_k = \begin{cases} \min(G_k/G_1, 1), & G_k > 0,\ G_1 > 0, \text{ and no nonfinite entries} \\ & \text{occur in the direction construction at state } k, \\ 0, & \text{otherwise.} \end{cases} \tag{S6}$$

Each variant's proposals determine its gains and weights under this shared rule.

Backpropagated targets.

- The *Euler weights (backpropagated targets)* variant fits bounded backpropagated targets with Euler coefficients.
- *Two half-steps, gain weights* takes two half-radius pullback steps, recomputes the direction after the first, and projects each proposal into the original ball before gain-weighted fitting.

Teacher supervision. *No teacher* removes distillation and its gradient-norm balancing while retaining the reference-similarity reward.

Paired comparisons. The tables below report FLUX.2-klein-4B ablations and paired comparisons, with one training realization per configuration and case. Paired effects use unrounded case-level differences and 20,000 bootstrap draws within the object and style groups, with equal group weights. The pointwise 95% intervals describe case variation, do not estimate variation across training seeds, and are not adjusted for multiple comparisons. Paired tables report the three reward-related metrics. The means tables and Table 2 give all metrics.

Color comparisons. Subscripts in Table 1 subtract displayed D-OPSD means within each model, while those in Table 2 subtract unrounded SRA means. Arrows give the sign of the difference; colors indicate improvement or deterioration without testing statistical significance.

Reward-state coverage. Panel A of Table S10 retains teacher supervision at every state and varies where reward feedback is applied. Distributing it across states improves DINO-D on reference rewrites, and concentrating it at the final prediction improves DINOv3 and Quality-S. All three paired intervals exclude zero. The LPIPS-D means in Table 2 move with DINO-D. Panel B additionally applies gain verification and reports a separate configuration comparison.

Table S9: **Construction and state-weight ablations on FLUX.2-klein-4B.** Means over twenty cases, one training run per configuration and case. All configurations retain teacher supervision and the same aggregate gradient-norm ratio. Uniform-weight ReFL repeats Table 1. Rows follow Table 2.

	Reward-related metrics			Other evaluation metrics			
Configuration	DINO-D↓	DINOv3↑	Quality-S↑	LPIPS-D↓	VLM-J↑	CLIP-S↑	Aesthetic-S↑
ReFL, uniform	0.1650	0.7343	0.7232	0.7224	1.9188	0.3327	5.5737
ReFL, Euler	0.1424	0.7462	0.7782	0.7204	2.0969	0.3304	5.6216
One-sided, uniform	0.1563	0.7405	0.7514	0.7218	1.9531	0.3314	5.6080
One-sided, Euler (SRA)	0.1417	0.7469	0.7793	0.7200	2.1250	0.3316	5.6057
Two-sided, uniform	0.1524	0.7427	0.7685	0.7200	1.9937	0.3306	5.6495
Two-sided, Euler	0.1362	0.7505	0.7983	0.7184	2.0187	0.3318	5.6445

Table S10: **Reward-state coverage on Z-Image-Turbo.** Means and paired differences. Both variants retain teacher supervision at all four states. Panel B is a separate configuration that fits the final target only after a positive endpoint gain. Differences are the first minus the second configuration in each metric's native direction, and arrows give the better direction. Brackets give pointwise 95% case-bootstrap intervals over twenty cases, with one training run per configuration and case.

Configuration	DINO-D↓	DINOv3↑	Quality-S↑
SRA	0.1503	0.7283	0.8382
A. Final-state-only			
Final-state-only	0.1554	0.7336	0.8479
SRA − final-state-only	−0.0051 [−0.0106, −0.0004]	−0.0053 [−0.0089, −0.0016]	−0.0097 [−0.0175, −0.0019]
B. Final-state-only with gain verification			
Gain-verified	0.1608	0.7273	0.8412
SRA − gain-verified	−0.0105 [−0.0159, −0.0060]	+0.0010 [−0.0036, +0.0056]	−0.0030 [−0.0089, +0.0027]

State weights. Uniform-versus-Euler comparisons vary relative state contributions within each reward construction at the same prescribed aggregate norm ratio (Table S11). Euler weighting improves the three reward-related metrics across the tested constructions on both models.

The Z-Image-Turbo last-heavy control retains reward supervision at all four states. Its weights $[0.2, 0.2, 0.2, 1]$ match the Euler weights $[1/10, 1/6, 1/3, 1]$ in the final state's explicit weight and the sum of the preceding weights (0.6). These matches do not fix the states' actual shares of the aggregate parameter gradient. Last-heavy improves all three reward-related metrics over uniform weighting. Compared with SRA, last-heavy improves Quality-S by 0.0056 [0.0007, 0.0103], and the DINO-D and DINOv3 intervals include zero (Table S11). These results support simple late-state emphasis in this setting without establishing the exact Euler weights as optimal or the two weighting rules as equivalent.

Table S11: **State weights across reward constructions.** Euler weights are compared with uniform weights within each construction at the same aggregate gradient-norm ratio. Differences are the first minus the second configuration in each metric's native direction, and arrows give the better direction. Brackets give pointwise 95% case-bootstrap intervals over twenty cases, with one training run per configuration and case.

Construction (Euler − uniform)	DINO-D↓	DINOv3↑	Quality-S↑
Z-Image-Turbo			
One-sided	−0.0184 [−0.0221, −0.0138]	+0.0187 [+0.0123, +0.0252]	+0.0551 [+0.0483, +0.0626]
Two-sided	−0.0222 [−0.0286, −0.0173]	+0.0187 [+0.0112, +0.0262]	+0.0581 [+0.0483, +0.0692]
ReFL	−0.0234 [−0.0310, −0.0164]	+0.0209 [+0.0145, +0.0278]	+0.0642 [+0.0585, +0.0701]
FLUX.2-klein-4B			
One-sided	−0.0146 [−0.0186, −0.0108]	+0.0064 [+0.0039, +0.0093]	+0.0279 [+0.0235, +0.0325]
Two-sided	−0.0162 [−0.0200, −0.0128]	+0.0079 [+0.0050, +0.0110]	+0.0299 [+0.0252, +0.0347]
ReFL	−0.0227 [−0.0286, −0.0171]	+0.0119 [+0.0083, +0.0157]	+0.0550 [+0.0482, +0.0618]

Local rescaling. With Euler weights fixed, SRA versus ReFL evaluates combined prediction-norm scaling and reward-gradient normalization. To separate the two factors, Table S12 reports a full 2×2 training ablation on both base models. Configuration A is Euler-weighted ReFL; B multiplies each Euler coefficient by the state's prediction norm divided by the mean prediction norm in that trajectory.

Table S11: **State weights (continued): last-heavy control on Z-Image-Turbo.** One-sided targets with weights $[0.2, 0.2, 0.2, 1]$ in rollout order, at the same aggregate gradient-norm ratio. The first row gives means. Differences are the first minus the second configuration in each metric's native direction, and arrows give the better direction. Brackets give pointwise 95% case-bootstrap intervals over twenty cases, with one training run per configuration and case.

Configuration	DINO-D↓	DINOv3↑	Quality-S↑
Last-heavy	0.1520	0.7309	0.8437
Last-heavy − uniform	−0.0166 [−0.0202, −0.0129]	+0.0212 [+0.0153, +0.0276]	+0.0606 [+0.0516, +0.0709]
Last-heavy − Euler (SRA)	+0.0018 [−0.0011, +0.0049]	+0.0026 [−0.0018, +0.0072]	+0.0056 [+0.0007, +0.0103]

Configuration C fits normalized one-sided targets with the common radius $\delta K^{-1} \sum_j \|\hat{x}_j^s\|_2$; D is SRA, with individual prediction-norm radii. The common radius can vary across trajectories. Within each base model, all four configurations retain its Euler weights, teacher supervision, prescribed aggregate gradient-norm ratio, and the twenty-case, 1,000-update protocol. For fixed predictions and shared reward gradients, under exact aggregate norm balancing, the relative local factors for A–D are respectively 1, $\|\hat{x}_k^s\|_2$, $1/\|g_k\|_2$, and $\|\hat{x}_k^s\|_2/\|g_k\|_2$. This identity has the numerical qualifications given in Appendix B; the training comparisons measure the resulting configurations rather than fixing gradients across their runs.

The normalization comparisons (C versus A and D versus B) improve all three reward-related metrics on Z-Image-Turbo, with all six paired intervals excluding zero. On FLUX.2-klein-4B, these differences are mostly unresolved, except for DINO-D without prediction scaling (C − A): -0.0036 $[-0.0081, -0.0010]$. The prediction-scaling comparisons (B versus A and D versus C) have intervals including zero on all three reward-related metrics on both models. Thus, the measured benefit of normalization depends on the backbone, while these comparisons do not establish an additional benefit from prediction-norm scaling on the three metrics, or equivalence between configurations.

The two-factor means in Table S12 also give the other evaluation metrics for all four configurations.

Table S12: **Local reward construction at fixed Euler weights.** All configurations retain teacher supervision and the same aggregate gradient-norm ratio. Differences are the first minus the second configuration in each metric's native direction, and arrows give the better direction. Brackets give pointwise 95% case-bootstrap intervals over twenty cases, with one training run per configuration and case.

Comparison	DINO-D↓	DINOv3↑	Quality-S↑
Z-Image-Turbo			
Two-sided, Euler − SRA	−0.0029 [−0.0074, +0.0008]	+0.0053 [−0.0002, +0.0107]	+0.0212 [+0.0140, +0.0304]
SRA − ReFL, Euler	−0.0088 [−0.0127, −0.0054]	+0.0054 [+0.0015, +0.0091]	+0.0269 [+0.0214, +0.0329]
FLUX.2-klein-4B			
Two-sided, Euler − SRA	−0.0054 [−0.0108, −0.0015]	+0.0036 [+0.0021, +0.0053]	+0.0190 [+0.0154, +0.0227]
SRA − ReFL, Euler	−0.0007 [−0.0017, +0.0004]	+0.0007 [−0.0008, +0.0022]	+0.0011 [−0.0001, +0.0024]

Reward directions and target construction. The backpropagated variants regress toward bounded targets built from normalized continuation-reward gradients. DRaFT directly differentiates the final-image reward. Per-state normalization and regression weights change how contributions combine (Appendix B), so these procedures need not have the same behavior. The remaining target variants in Table 2 are compared descriptively by their means. With Euler weights, local SRA targets improve DINO-D and Quality-S over backpropagated targets, while the latter have higher Aesthetic-S. Under the shared gain-weighting rule, random targets have reward-related means close to D-OPSD. Two-sided gain-weighted targets improve DINOv3 and Quality-S over SRA, while worsening both DINO-D and LPIPS-D. The two-half-step variant improves all three reward-related means over one gain-weighted pullback step, while SRA remains better on these metrics. Table S13 gives paired comparisons for full-trajectory DRaFT.

Teacher supervision and aggregate strength. *No teacher* removes both the distillation loss and the balancing rule that uses its gradient, so it evaluates their joint removal. All three reward-related means improve while the other four worsen. The κ sweep retains teacher supervision and varies relative reward strength, with $\kappa = 0$ being D-OPSD. For SRA on Z-Image-Turbo (Table 3), the reward-related metrics improve monotonically with κ and do not plateau by 0.6, whereas CLIP-S

Table S12: **Local reward construction (continued): two-factor means.** N denotes local reward-gradient normalization and P prediction-norm scaling (0: off, 1: on). A: Euler-weighted ReFL. B: prediction-norm-weighted Euler-weighted ReFL. C: common-radius normalized targets. D: SRA. All configurations use Euler weights, teacher supervision, and the same aggregate gradient-norm ratio within each base model, with twenty cases and one training run per configuration and case.

	Reward-related metrics			Other evaluation metrics			
Config. (N,P)	DINO-D↓	DINOv3↑	Quality-S↑	LPIPS-D↓	VLM-J↑	CLIP-S↑	Aesthetic-S↑
Z-Image-Turbo							
A (0,0)	0.1591	0.7229	0.8113	0.7197	1.7375	0.3182	5.6278
B (0,1)	0.1560	0.7231	0.8152	0.7195	1.7469	0.3182	5.6213
C (1,0)	0.1506	0.7291	0.8374	0.7210	1.8187	0.3179	5.6336
D (1,1)	0.1503	0.7283	0.8382	0.7218	1.7625	0.3191	5.6285
FLUX.2-klein-4B							
A (0,0)	0.1424	0.7462	0.7782	0.7204	2.0969	0.3304	5.6216
B (0,1)	0.1423	0.7463	0.7791	0.7206	2.0812	0.3310	5.6157
C (1,0)	0.1388	0.7466	0.7793	0.7208	2.1000	0.3316	5.6117
D (1,1)	0.1417	0.7469	0.7793	0.7200	2.1250	0.3316	5.6057

Table S12: **Local reward construction (continued): the two factors.** Each row switches on one factor at a fixed setting of the other: N rows switch on gradient normalization and P rows prediction-norm scaling. Configurations A–D are defined above. Differences are the first minus the second configuration in each metric's native direction, and arrows give the better direction. Brackets give pointwise 95% case-bootstrap intervals over twenty cases, with one training run per configuration and case.

Comparison	DINO-D↓	DINOv3↑	Quality-S↑
Z-Image-Turbo			
N: C − A	−0.0084 [−0.0118, −0.0051]	+0.0063 [+0.0027, +0.0100]	+0.0261 [+0.0205, +0.0325]
N: D − B	−0.0057 [−0.0116, −0.0010]	+0.0052 [+0.0010, +0.0092]	+0.0230 [+0.0156, +0.0317]
P: B − A	−0.0031 [−0.0065, +0.0007]	+0.0002 [−0.0034, +0.0036]	+0.0039 [−0.0054, +0.0124]
P: D − C	−0.0004 [−0.0023, +0.0018]	−0.0009 [−0.0047, +0.0031]	+0.0008 [−0.0021, +0.0038]
FLUX.2-klein-4B			
N: C − A	−0.0036 [−0.0081, −0.0010]	+0.0004 [−0.0008, +0.0016]	+0.0011 [−0.0004, +0.0027]
N: D − B	−0.0006 [−0.0015, +0.0003]	+0.0006 [−0.0006, +0.0016]	+0.0002 [−0.0012, +0.0015]
P: B − A	−0.0001 [−0.0011, +0.0013]	+0.0001 [−0.0012, +0.0013]	+0.0010 [−0.0008, +0.0029]
P: D − C	+0.0029 [−0.0003, +0.0082]	+0.0003 [−0.0007, +0.0012]	+0.0000 [−0.0011, +0.0012]

Table S13: **Euler weighting in full-trajectory DRaFT on Z-Image-Turbo.** All configurations retain the same teacher supervision and aggregate gradient-norm ratio. Differences are the first minus the second configuration in each metric's native direction, and arrows give the better direction. Brackets give pointwise 95% case-bootstrap intervals over twenty cases, with one training run per configuration and case.

Comparison	DINO-D↓	DINOv3↑	Quality-S↑
DRaFT, Euler − DRaFT	−0.0212 [−0.0291, −0.0138]	+0.0207 [+0.0146, +0.0268]	+0.0774 [+0.0651, +0.0908]
SRA − DRaFT, Euler	−0.0090 [−0.0142, −0.0044]	+0.0013 [−0.0058, +0.0083]	+0.0057 [−0.0023, +0.0142]

decreases monotonically and its difference from D-OPSD exceeds the case-bootstrap interval only at $\kappa = 0.6$ (-0.0091, interval $[-0.0143, -0.0038]$). Aesthetic-S peaks between 0.1 and 0.3. Table S14 reports the same sweep on FLUX.2-klein-4B. Reward-related metrics again improve and CLIP-S decreases as κ increases. VLM-J and Aesthetic-S have their largest tested means at 0.3 on this model, whereas Z-Image-Turbo's mean VLM-J continues to increase at 0.6. The reward-related ordering of SRA against D-OPSD holds at all tested positive ratios on both models, and we retain 0.3 as an intermediate balance between reward gain and prompt-adherence cost. On Z-Image-Turbo, Table S15 compares SRA and Euler-weighted ReFL at each of the three positive ratios, using the same Euler state weights, teacher supervision, twenty cases, and shared training protocol. The prescribed ratio matches aggregated raw parameter gradient norms before clipping and AdamW preconditioning. All nine reward-related point estimates favor SRA, and eight pointwise 95% case-bootstrap intervals exclude zero. The exception is DINO-D at $\kappa = 0.6$: -0.0050 $[-0.0114, 0.0002]$ (Table S15). These comparisons extend the reward-related gains over Euler-weighted ReFL beyond the default $\kappa = 0.3$. CLIP-S decreases with κ in both constructions, and at each ratio the two constructions' CLIP-S means differ by at most 0.0013. This does not establish equal prompt alignment or a lower prompt-alignment cost for SRA.

Table S14: **Aggregate reward strength on FLUX.2-klein-4B.** Means over twenty cases with all other settings fixed, one training run per configuration and case.

	Reward-related metrics			Other evaluation metrics			
Configuration	DINO-D↓	DINOv3↑	Quality-S↑	LPIPS-D↓	VLM-J↑	CLIP-S↑	Aesthetic-S↑
$\kappa = 0$ (D-OPSD)	0.1954	0.7207	0.5306	0.7242	1.7875	0.3354	5.2513
$\kappa = 0.1$	0.1683	0.7308	0.7061	0.7238	1.9375	0.3347	5.5423
$\kappa = 0.3$ (SRA)	0.1417	0.7469	0.7793	0.7200	2.1250	0.3316	5.6057
$\kappa = 0.6$	0.1204	0.7579	0.8311	0.7234	2.0375	0.3300	5.5907

Table S15: **Matched reward strength on Z-Image-Turbo: means.** Euler-weighted ReFL and SRA share the Euler state weights and teacher supervision at each reward-to-distillation gradient-norm ratio κ. $\kappa = 0$ is D-OPSD. Twenty cases, one training run per configuration and case.

	Reward-related metrics			Other evaluation metrics			
Configuration	DINO-D↓	DINOv3↑	Quality-S↑	LPIPS-D↓	VLM-J↑	CLIP-S↑	Aesthetic-S↑
$\kappa = 0$, D-OPSD	0.2278	0.6792	0.6408	0.7216	1.4531	0.3227	5.3031
$\kappa = 0.1$, ReFL, Euler	0.1827	0.7019	0.7568	0.7220	1.6094	0.3211	5.5914
$\kappa = 0.1$, SRA	0.1783	0.7058	0.7738	0.7225	1.5656	0.3210	5.6182
$\kappa = 0.3$, ReFL, Euler	0.1591	0.7229	0.8113	0.7197	1.7375	0.3182	5.6278
$\kappa = 0.3$, SRA	0.1503	0.7283	0.8382	0.7218	1.7625	0.3191	5.6285
$\kappa = 0.6$, ReFL, Euler	0.1402	0.7340	0.8472	0.7229	1.8062	0.3149	5.5639
$\kappa = 0.6$, SRA	0.1351	0.7455	0.8719	0.7253	1.9219	0.3136	5.5565

Table S15: **Matched reward strength (continued): paired differences.** Each row gives SRA minus Euler-weighted ReFL at the same κ. Differences are the first minus the second configuration in each metric's native direction, and arrows give the better direction. Brackets give pointwise 95% case-bootstrap intervals over twenty cases, with one training run per configuration and case.

Ratio	DINO-D↓	DINOv3↑	Quality-S↑
$\kappa = 0.1$	-0.0045 $[-0.0069, -0.0020]$	$+0.0039$ $[+0.0013, +0.0069]$	$+0.0170$ $[+0.0130, +0.0213]$
$\kappa = 0.3$	-0.0088 $[-0.0127, -0.0054]$	$+0.0054$ $[+0.0015, +0.0091]$	$+0.0269$ $[+0.0214, +0.0329]$
$\kappa = 0.6$	-0.0050 $[-0.0114, +0.0002]$	$+0.0115$ $[+0.0023, +0.0217]$	$+0.0248$ $[+0.0131, +0.0375]$

Fixed-checkpoint gradient probes. At fixed checkpoints and matched rollout coordinates, we compare the auxiliary parameter gradients of configurations A–D and of uniform-weight ReFL at three probe positions, before balancing, teacher-gradient addition, clipping, and AdamW. On FLUX.2-klein-4B, Euler-weighted ReFL and SRA are closely aligned, with mean cosines of 0.96

to 0.99 over two cases and two checkpoints, whereas uniform-weight ReFL is less aligned (0.41 to 0.66). This is consistent with the small training-level differences between A and D on this backbone, without establishing performance equivalence. With reward gradients cached and shared, the rescaling predicted in Appendix B matches actual target-fitting backpropagation, with cosines of at least 0.9997 and relative errors of at most 2.2% after the $1/N$ factor. Separately evaluated reward gradients at identical anchors can, however, differ in direction. On one Z-Image-Turbo case, the four per-state cosines between two evaluations are 0.998, 0.975, 0.823, and 0.765 in rollout order, and repeated calls at the final state give pairwise cosines as low as 0.514, or 0.81 to 0.96 with the deterministic settings of Appendix D.4. Local normalization removes raw magnitude from target length but retains this sensitivity to direction. These diagnostics locate the variability without identifying its cause, and the training gains above are not attributed to it.

F Reference-Free Preference and Quality Scorers

Appendices C and E define the main metrics, their reward membership, and the shared scoring protocol.

Additional scorers. This appendix reports six additional preference and quality scorers on the configurations in Table S16, including ReFL (uniform). Four are preference models conditioned on the evaluation prompt: PickScore (Kirstain et al., 2023), ImageReward (Xu et al., 2023), HPSv2.1 (Wu et al., 2023), and HPSv3 (Ma et al., 2025b). Two are no-reference quality and aesthetic predictors that see only the image: DeQA-Score (You et al., 2025) and the LAION aesthetic predictor v2 (Schuhmann et al., 2022). None of the six takes the reference image as input. All six scorers are frozen and none is used in the main DINOv3–TOPIQ training reward. PickScore and ImageReward are used in the alternative reward recipes of Appendix C.1. They are scored on the same sixteen evaluation-prompt images per case as VLM-J, with the reduction of Appendix E.

Table S16: **Reference-free preference and quality scorers on the evaluated configurations.** Mean over the twenty cases for each base model, on the sixteen evaluation-prompt images per case. Rows, checkpoints, and images are those of Table 1. Bold and underline denote the best and second-best values within each model. None of the six scorers takes the reference image as input. Appendix F discusses their relationship to reference fidelity and the main quality and aesthetic scores.

Method	PickScore↑	ImageReward↑	HPSv2.1↑	HPSv3↑	DeQA↑	LAION-Aes↑
Z-Image-Turbo						
Base model, no adaptation	**22.9823**	**1.3330**	**0.2975**	**11.1558**	**4.2242**	**5.9641**
SFT	20.9339	0.2293	0.2240	3.9986	2.9918	5.3167
DreamBooth	22.1527	1.0355	0.2660	8.9611	3.4966	5.6951
PSO	22.2503	0.9017	0.2645	8.9085	3.3974	5.7392
SFT → LoRA	22.5041	1.1642	0.2874	10.7917	4.1053	5.8637
D-OPSD	22.3596	1.0018	0.2749	9.7466	3.9857	5.9272
+ ReFL (uniform)	22.1883	0.9211	0.2745	9.6019	4.0125	5.9467
+ DRaFT	22.2027	0.9298	0.2750	9.6363	4.0064	5.9261
+ SRA	21.9413	0.7937	0.2706	9.1012	3.8720	5.9066
FLUX.2-klein-4B						
Base model, no adaptation	**23.4866**	**1.5703**	**0.3139**	**11.3466**	**4.1649**	**6.1784**
SFT	20.7844	0.2590	0.2101	2.9426	2.7403	5.3778
DreamBooth	21.4585	0.6338	0.2299	4.7483	2.8920	5.5868
PSO	22.5956	1.1677	0.2821	8.4960	3.1690	5.8976
SFT → LoRA	23.1863	1.5202	0.3091	11.0422	4.0614	6.0746
D-OPSD	22.2685	1.0316	0.2813	9.5253	3.5494	5.8714
+ ReFL (uniform)	22.1640	1.0143	0.2813	9.7275	3.6790	5.9130
+ DRaFT	22.2069	0.9978	0.2815	9.7028	3.7070	5.9245
+ SRA	21.9936	0.9179	0.2748	9.3152	3.6516	5.8777

Reading. In Table S16 the unadapted base model has the highest value in every column on both base models. Every adapted method is below it, including SFT and DreamBooth, which train under no reward at all. Among the D-OPSD variants evaluated in this panel, SRA is lowest on the four prompt-conditioned preference models. The case-paired difference between SRA and D-OPSD has a bootstrap interval covering zero on LAION-Aes for both base models and on DeQA for Z-Image-Turbo, and SRA is above D-OPSD on DeQA for FLUX.2-klein-4B by 0.10 with a bootstrap interval of $[0.02, 0.19]$.

Preference scores and reference fidelity. The human judgments of Appendix G, made with the reference visible, favor SRA, and they do not identify why the two protocols disagree. The following measurements relate preference and quality scores to reference fidelity.

First, the adaptation target itself scores below the base model's output. On Z-Image-Turbo we scored each case's reference photograph with that case's rewrite prompt, beside the rewrite images every method generated from the same prompt. PickScore rates the reference photographs at 20.53 on average and the base model's images at 21.64, and it places the base model above the reference on 16 of 20 cases. D-OPSD and SRA read 21.04 and 21.06, roughly half way between the two. This illustrates that reference fidelity and PickScore assess different objectives.

Second, PickScore reductions are associated with reference-similarity gains across cases. For D-OPSD and its three reward variants, on both base models, the per-case change in PickScore against the base model correlates negatively with the per-case change in DINOv3, with Pearson coefficients between -0.32 and -0.56, and DreamBooth reads -0.41 and -0.63. Across methods, SFT $\rightarrow$ LoRA ranks second on all four prompt-conditioned preference scores on both base models in Table S16. DreamBooth, which uses no reward, loses more PickScore than D-OPSD on both base models (-0.83 against -0.62 on Z-Image-Turbo and -2.03 against -1.22 on FLUX.2-klein-4B).

Third, the main quality and aesthetic scores increase in the same rewrite-prompt comparison. Under the same reference-photograph reading, Aesthetic-S rates the reference photographs at 4.57 and the base model's images at 5.29, yet SRA reaches 5.63 and is above the reference on 20 of 20 cases. Quality-S reads 0.34, 0.63, and 0.86 on the same three sets. These measurements show lower PickScore alongside higher TOPIQ and MUSIQ scores for SRA relative to the base model in this comparison.

The relative scores also depend on the prompt panel. On Z-Image-Turbo's rewrite prompt, which is the training prompt, SRA and D-OPSD have similar mean PickScore, as reported above, whereas SRA scores lower on the re-contextualized evaluation prompts in Table S16.

G Supplementary Human Preference Study

We collected human judgments with both the reference and the prompt visible, alongside the automatic evaluations in Appendix F. The study provides supplementary descriptive evidence on the development validation set. The analysis of the responses was conducted after interim aggregates had been inspected.

The study compares SRA against D-OPSD and against DRaFT on both base models, using the checkpoints and evaluation images of Table 1. A fixed seed draws one evaluation-prompt image per reference row for each of the twenty cases, at the same case, reference row, prompt, and generation seed for every method, giving $2 \times 2 \times 20 \times 4 = 320$ pairs whose two images differ only in the method. Raters see the reference image, the prompt, and the two generated images in randomized left and right positions, with no method names, and answer one question, overall preference, by choosing the left image, the right image, "about the same", or "cannot judge". Seven participants completed 11 anonymous links of 40 pairs, submitting 440 judgments that cover 275 of the 320 pairs and all twenty cases in every row of Figure 5. No rater chose "about the same" or "cannot judge". Which participant completed which link was not recorded, so the link is the unit of analysis and the number of links judging a pair bounds the number of distinct people judging it from above.

A judgment for the SRA image scores 1 and against it 0. Judgments are averaged within a pair, pairs within a case, and cases with equal weight. Unrated pairs are omitted, never imputed. The whiskers in Figure 5 are 95% percentile bootstraps over the observed cases, stratified by object and style (10,000 draws), and describe case variation under the observed raters, images, and coverage only.

Preference for SRA is near unanimous on Z-Image-Turbo's object cases and lowest on style cases, consistent with the style construct caveat of Appendix E. Dropping any one link and recomputing moves each all-case rate by at most four points, with all twenty cases retained. This check cannot detect dependence on a participant who completed several links. A single overall preference cannot separate reference fidelity, prompt adherence, and image quality.

H Training and Validation Curves

The ReFL and two-sided configurations in this checkpoint study both use uniform state weights. Their curves and training times refer to these configurations throughout this section.

Training rewards. Figures 3(a, b) and S1(a) use each method's training-chain rewards over twenty cases. For each comparator, we average the paired reward difference over the final 100 updates within each case, then take the median across cases. These median gains are 0.74 over the two-sided construction, 1.96 over DRaFT, and 1.98 over ReFL. Every comparison favors SRA in all twenty cases. The figure's bootstrap intervals resample cases and describe uncertainty in the group mean or paired median. Across-case standard deviations at update 1,000 range from approximately 0.8 to 1.3, wider than the intervals for the means. D-OPSD loads no reward during training and therefore has no curve.

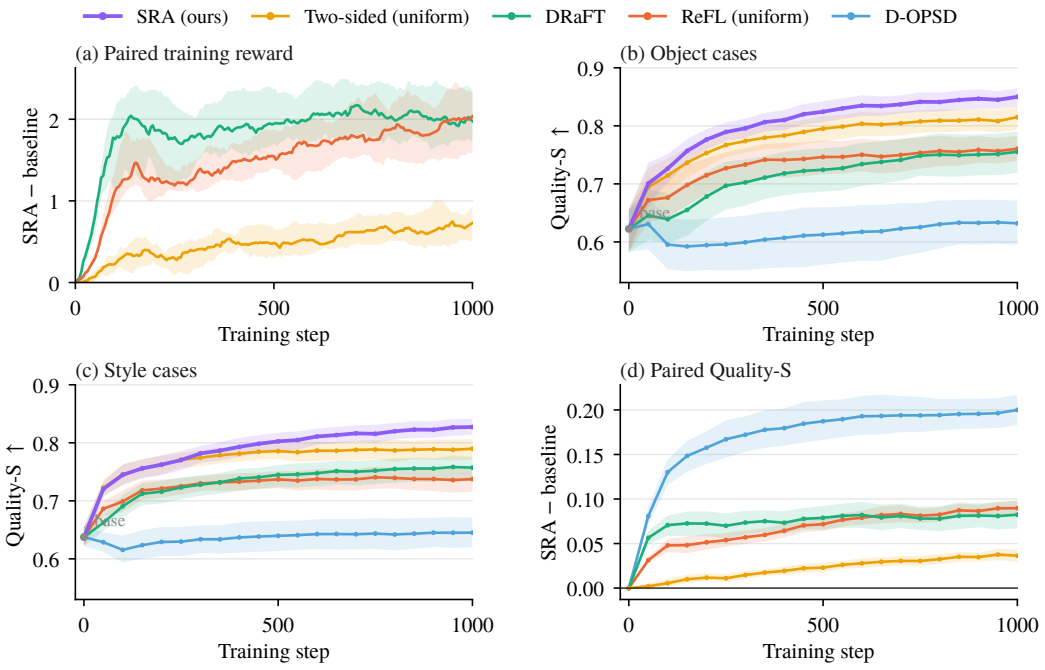


Figure S1: **Training and validation curves against update count.** (a) Median paired training reward gap to SRA. (b, c) Mean validation Quality-S over ten object and ten style cases, scored every 50 updates. (d) Paired Quality-S difference to SRA over all twenty cases. Shading is 95% case bootstrap. Grey dots mark the base model.

Validation checkpoints and training time. The checkpoint study trains each of the five methods for 1,000 updates from the same base model on all twenty cases. It saves the student adapter every 50 updates and scores each checkpoint on the case's standard validation panel. Step 0 is the unadapted base model. Figure 3(d–f) plots the three reward-related metric means against cumulative training time on one H200 GPU, excluding scoring passes. In Figure 3(c), the targets are each comparator's final mean DINO-D, DINOv3, and Quality-S from this same study. We use the first saved SRA checkpoint reaching all three means simultaneously (lower DINO-D, higher DINOv3 and Quality-S), without interpolation, and divide the comparator's final mean training time by SRA's mean training

time at that checkpoint. Ratios are computed before rounding. These thresholds compare observed case means and do not require every case to cross its own comparator's thresholds. Considering Quality-S alone, SRA reaches the final means of DRaFT and ReFL at the 150-update checkpoint, costing about 0.32 GPU hours against 3.26 and 2.05 hours for their 1,000 updates, and the final Quality-S of the two-sided construction at 450 updates (0.95 against 2.68 hours). Each method has one training run per case. The intervals resample cases and do not estimate variation across training replicates. The time ratios are descriptive point estimates without confidence intervals. Crossing observed mean thresholds does not establish statistical equivalence or matching performance on the other evaluation metrics. Figure S2 reports DINO-D at the same checkpoints, using four reference rewrites per case.

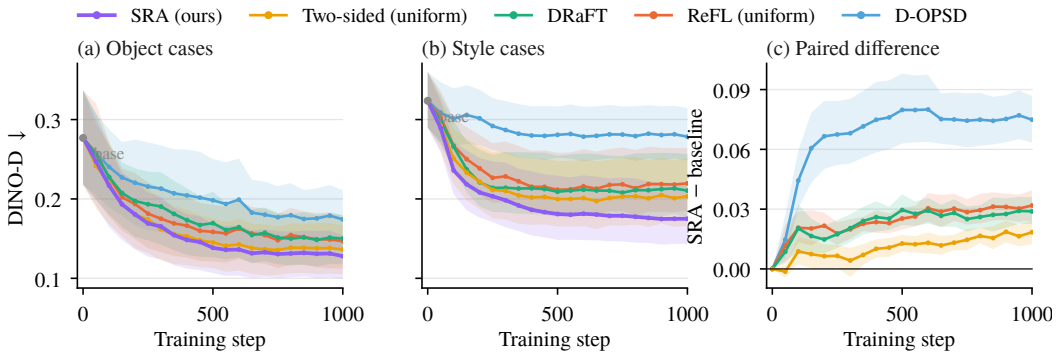


Figure S2: **DINO-D over the same validation checkpoints.** (a, b) Mean over the ten object and ten style cases, (c) paired difference to SRA over the twenty cases. Shading is the 95% case-resampled bootstrap.

I Limitations

Our method requires a base objective and a differentiable reward. We evaluate SRA with reference-conditioned D-OPSD. Its effectiveness with other base objectives remains untested. The main experiments use a weighted composite of DINOv3 reference similarity and TOPIQ image quality. Appendix C.1 also evaluates a DINOv3–PickScore composite and single-term DINOv3, TOPIQ, and ImageReward recipes. The main reward does not explicitly measure prompt adherence and can disagree with an independent judge. Shared parameters, the distillation gradient, and optimizer preconditioning separate a target's gain from fitting success. The fixed-model diagnostics therefore establish neither deployment quality nor a computational advantage. Those require evaluation of the fitted student under the deployment protocol.

Local normalization remains sensitive to reward-gradient direction errors (Appendix E.4). Their prevalence across models and training remains unmeasured; fixed-checkpoint probes cannot substitute for complete training comparisons.

J Additional Qualitative Comparisons

Figures S3 and S4 extend Figure 4 to both base models.

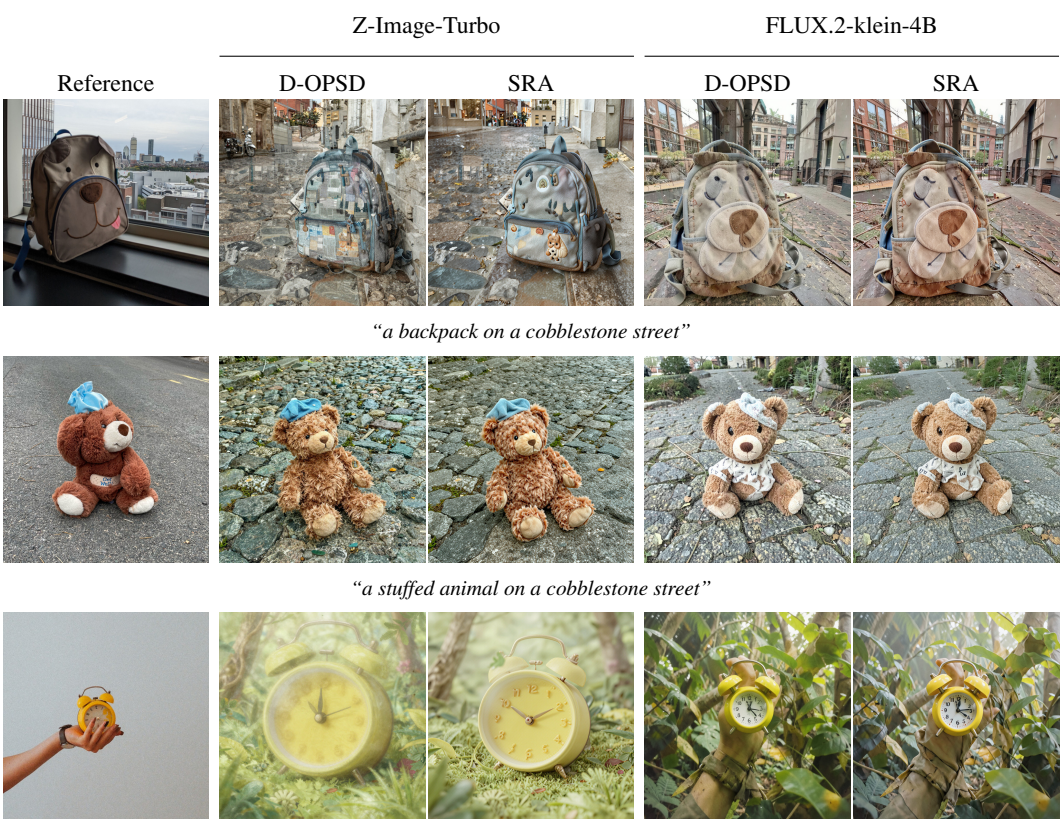


Figure S3: **Additional object comparisons.** Each row shows the reference and D-OPSD and SRA on Z-Image-Turbo and FLUX.2-klein-4B. All generated images in a row share the prompt and the seed. On both base models, SRA renders the subject more cleanly.

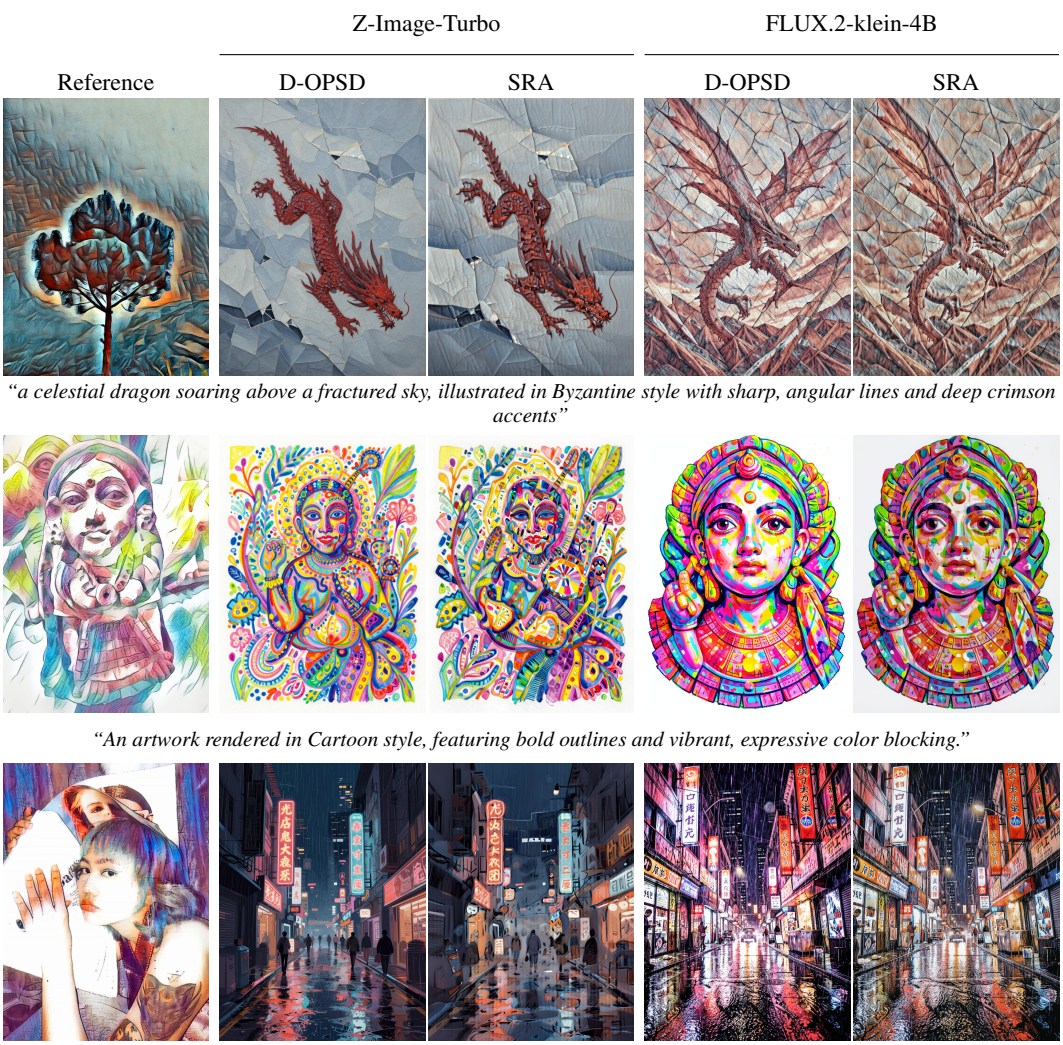


Figure S4: **Additional style comparisons.** Rows follow Figure S3. On both base models, SRA carries more of the reference's brushwork, visible as faceted relief in the Byzantine row, fragmented color blocks in the Cartoon row, and smeared strokes in the Cold Warm row.