Drifting Preference Optimization for One-Step Generative Models

Zhou Jiang[1] **Yandong Wen**[1] **Zhen Liu**[2*]
[1]Westlake University [2]The Chinese University of Hong Kong, Shenzhen

Abstract

One-step text-to-image generators are attractive for deployment because they generate an image with a single forward pass, but preference finetuning them remains difficult: standard alignment methods often rely on policy likelihoods, denoising trajectories, differentiable reward gradients, or test-time optimization. We propose Drifting Preference Optimization (DrPO), an online preference-finetuning method for deterministic one-step generators. For each prompt, DrPO samples candidates from the current generator, ranks them with a target reward, and uses high- and low-scoring samples to synthesize a feature-space update direction. The update is a non-parametric dipole preference field plus a reference drift estimated from the frozen base generator, and is optimized through a detached feature-space regression target. The target reward is used only for ranking, so DrPO can train with large, black-box, or non-differentiable rewards while inference remains a single generator call. We evaluate DrPO on SD-Turbo and SDXL-Turbo with multiple target rewards and benchmarks, including HPSv3 and GenEval. DrPO improves alignment over reward-gradient-free one-step preference baselines and reduces HPSv3 training computation by $3.51\times$ under the matched effective-batch setting by removing reward-model backpropagation. Initial offline experiments suggest that sample-based gradient synthesis can also be used beyond online reward ranking. Code is available here.

1 Introduction

Recent text-to-image systems increasingly target low-latency generation through one-step generative training [Deng et al., 2026, Geng et al., 2025, Sauer et al., 2023] and diffusion distillation [Song et al., 2023, Luo et al., 2023, Sauer et al., 2024, Yin et al., 2024b]. Compared with multi-step diffusion samplers, these models generate an image with a single forward pass, making them useful for interactive and resource-constrained deployment. Deployment, however, also requires generators to produce images that better match human preferences and task-specific criteria. Following the success of reinforcement learning from human feedback (RLHF) in language and diffusion models, it is natural to post-train one-step generators with reward models so that their outputs better align with such preferences.

Reward-based post-training is nontrivial for one-step generators. Standard RLHF and preference-optimization methods typically rely on stochastic policy probabilities, density ratios, or denoising trajectories to construct policy updates. Directly backpropagating a differentiable reward through the generated image avoids these requirements and can be effective for lightweight rewards, but it follows the local gradient of the reward model and requires differentiating through that reward. This becomes costly for large VLM-based rewards and impossible for black-box or discrete evaluators. We therefore seek a reward-driven, reference-aware update direction that can be recovered from samples and reward evaluations alone.

*Corresponding Author.

Preprint.

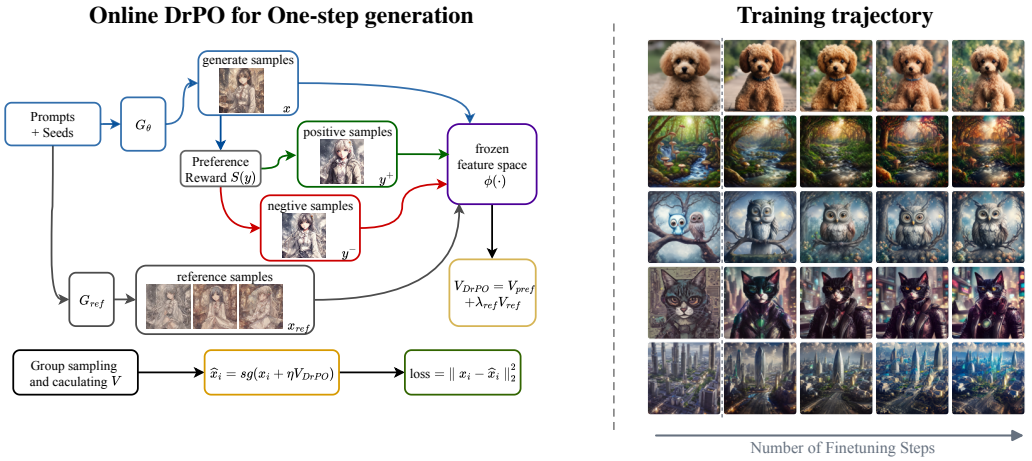


Figure 1: **Overview of online DrPO for one-step generation.** Left: the current generator samples candidate images for each prompt, and the target reward ranks them into preferred and rejected pairs. A frozen feature extractor maps these images into latent feature space, where preference pairs define the dipole drift and samples from the frozen base generator provide the reference correction. The combined drift gives a detached target for training the generator. Right: qualitative samples from fixed test prompts along the online finetuning trajectory.

We take inspiration from Drifting Models [Deng et al., 2026], which train one-step generators by constructing feature-space drift fields from finite batches of samples. In preference finetuning, reward-ranked candidates naturally define such a field: high-scoring samples provide attractive fields, while low-scoring samples provide repulsive fields. DrPO turns these fields into a non-parametric dipole preference field and independently estimates a reference drift from samples of the frozen base generator. The combined field gives a detached feature-space target for updating the current generator. Since the target reward is used only to rank generated candidates, DrPO supports expensive, black-box, or discrete rewards while preserving one-step inference. We evaluate DrPO across multiple reward models and one-step generators built on SD1.5 and SDXL. On HPSv3 and GenEval, DrPO improves preference alignment without differentiating through the target reward. We also report ablations on candidate count, feature choice, kernel choice, velocity scale, and reference regularization, along with initial results on offline preference finetuning.

Our core contributions are:

- We propose DrPO, an online preference-finetuning method that converts reward-ranked on-policy samples into a detached drifting update for one-step generators while keeping inference to one forward pass.
- We introduce a dipole preference field that builds latent-space update directions from positive and negative samples, replacing target-reward gradients with sample-based drifting updates.
- We validate DrPO on SD-Turbo and SDXL-Turbo across multiple rewards, where it gives the largest gains among reward-gradient-free one-step preference baselines.
- We show that DrPO works with large and non-differentiable rewards, including HPSv3 and GenEval-style rewards, and provide preliminary results on offline preference finetuning.

2 Related Work

One-step generative models. Fast image generators often compress iterative sampling into one or a few network evaluations through diffusion distillation [Yin et al., 2024b,a], consistency training [Lu and Song, 2024], adversarial objectives, flow distillation [Nguyen and Tran, 2024], or related mechanisms. Sampling then requires only a single forward pass, but many structures used by standard alignment methods are no longer available, including denoising trajectories, timestep-wise scores, sampler likelihoods, and tractable policy log probabilities. Existing fast-diffusion alignment methods often retain some diffusion-specific structure during training, for example through score regularization, pairwise margins, or step-level win–lose construction [Luo, 2025, Luo et al., 2025a, Miao et al., 2025, Croitoru et al., 2025, Luo et al., 2025b,c].

Preference alignment. RLHF aligns generative policies by learning a reward model from comparisons and updating a policy with PPO-style objectives under reference or KL control [Stiennon et al., 2020, Schulman et al., 2017, Ouyang et al., 2022]. DPO and related objectives remove the explicit RL loop by fitting likelihood-ratio preferences against a reference policy [Rafailov et al., 2023]; RLAIF and group-relative variants change the feedback source or normalization scheme while retaining a policy-optimization view [Bai et al., 2022, Shao et al., 2024]. In image generation, reward-gradient and preference methods adapt these ideas to diffusion or sampler structure: DRaFT backpropagates differentiable rewards through a sampler [Clark et al., 2024], diffusion-DPO adapts preference margins to diffusion likelihood bounds [Wallace et al., 2024], and ReNO or PAHI optimize sampling variables or prompt-specific noise distributions at test time [Eyring et al., 2024, Kim et al., 2024]. Recent preference-guidance methods also cast diffusion alignment as classifier-free-guidance-style inference [Jiang et al., 2026]. DrPO differs by estimating the update direction directly in feature space from reward-ranked samples and reference samples, without reward backpropagation, policy likelihoods, denoising trajectories, or test-time search.

Drifting models. Drifting Models introduce a likelihood-free training mechanism for one-step generators, where finite-batch drift in feature space replaces explicit density estimation or adversarial discrimination [Deng et al., 2026]. In their original formulation, the drift is used for distribution matching between generated samples and data samples. We use the same drifting view for reward-based post-training: the drift is defined by condition-specific preference information, while a frozen reference generator supplies the stability direction.

3 Preliminaries

3.1 Drifting Models

Drifting Models [Deng et al., 2026] train one-step generators by assigning a feature-space drift direction to each generated sample. For a condition c and noise ϵ, let $y_\theta = G_\theta(\epsilon, c)$ and let $f_\theta(\epsilon, c)$ be its feature representation. For a generated feature x, the drift moves the sample toward positive anchors and away from negative anchors:

$$\mathbf{V}_{\nu^+,\nu^-}(x) = \mathbb{E}_{b^+\sim\nu^+}\left[k(x,b^+)(b^+ - x)\right] - \mathbb{E}_{b^-\sim\nu^-}\left[k(x,b^-)(b^- - x)\right], \tag{1}$$

where k measures feature similarity, ν^+ denotes the positive distribution, and ν^- denotes the negative distribution. Given this field, the generator is trained against a detached drifted feature target:

$$\mathcal{L}_{\text{drift}}(\theta) = \frac{1}{2}\mathbb{E}_{\epsilon,c}\left[\left\| f_\theta(\epsilon,c) - \text{sg}\big(f_\theta(\epsilon,c) + \mathbf{V}_{\nu^+,\nu^-}(f_\theta(\epsilon,c))\big)\right\|_2^2\right], \tag{2}$$

where $\text{sg}(\cdot)$ denotes stop-gradient. Since the target is detached, this loss gives

$$\nabla_\theta \mathcal{L}_{\text{drift}} = -\mathbb{E}_{\epsilon,c}\left[\mathbf{V}_{\nu^+,\nu^-}(f_\theta(\epsilon,c))\nabla_\theta f_\theta(\epsilon,c)\right]. \tag{3}$$

Thus the drifting loss can be read as a deterministic policy-gradient update whose feature-space velocity is synthesized from samples.

For minibatch training, Drifting Models approximate the drift field with mean shift. Given a support batch $\mathcal{B} = \{b_i\}_{i=1}^B$, kernel weights are normalized within the batch:

$$w_i(x;\mathcal{B}) = \frac{k(x,b_i)}{\sum_j k(x,b_j)}, \qquad \widehat{\mu}_{\mathcal{B}}(x) = \sum_i w_i(x;\mathcal{B})b_i. \tag{4}$$

With positive and negative batches $\mathcal{B}^+$ and $\mathcal{B}^-$, the approximate drift field is

$$\widehat{\mathbf{V}}(x) = \widehat{\mu}_{\mathcal{B}^+}(x) - \widehat{\mu}_{\mathcal{B}^-}(x). \tag{5}$$

With an RBF kernel $k(x,b) = \exp(-s(x,b)/\tau)$, w_i is the softmax weight over negative similarities.

3.2 Reinforcement Learning from Human Feedback

RLHF first trains a reward model from pairwise preferences. Given comparisons (c, y^+, y^-), where y^+ is preferred to y^- under condition c, the reward model $r_\psi(c,y)$ is commonly trained with

$$\mathcal{L}_{\text{RM}}(\psi) = -\mathbb{E}_{(c,y^+,y^-)}\log\sigma\big(r_\psi(c,y^+) - r_\psi(c,y^-)\big). \tag{6}$$

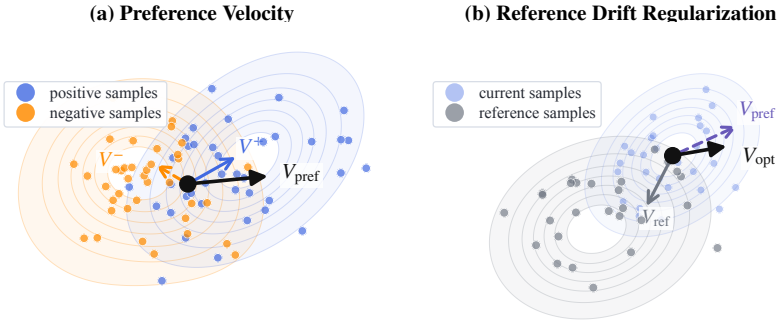


Figure 2: **Drift construction in DrPO.** (a) Reward-ranked positive and negative samples define the preference drift V_{pref}, which moves generated features toward preferred samples and away from rejected samples. (b) The final update direction combines this preference drift with the reference correction estimated from base and current samples, $V_{\text{DrPO}} = V_{\text{pref}} + \lambda V_{\text{ref}}$.

After reward learning, a common reference-regularized target is the tilted distribution

$$q^*(y \mid c) = q_{\text{ref}}(y \mid c) \exp(r_\psi(c, y)/\beta_{\text{KL}}) \, / Z(c), \tag{7}$$

where $Z(c)$ is the normalizing constant. Fitting q_θ to this target is equivalent to the soft RL objective [Haarnoja et al., 2018]

$$\max_{q_\theta} \; \mathbb{E}_c \mathbb{E}_{y \sim q_\theta(\cdot|c)} \left[r_\psi(c, y) \right] - \beta_{\text{KL}} \mathbb{E}_c \left[\text{KL}(q_\theta(\cdot \mid c) \,\|\, q_{\text{ref}}(\cdot \mid c)) \right]. \tag{8}$$

This formulation contains the two forces used in preference alignment: reward improvement and reference control. For deterministic one-step generators, policy likelihoods and density ratios are not available in closed form, motivating sample-based estimates of the corresponding update direction.

4 Method

4.1 Dipole Reward Model

Let $x = g_\theta(\epsilon, c)$ and $z = \phi(x)$, where ϵ is the input noise, x is the generated image, and ϕ is a fixed feature extractor. Given positive and negative feature samples $\mathcal{A}^+$ and $\mathcal{A}^-$, we define a non-parametric reward in feature space:

$$R_{\text{dipole}}(z) = \exp(E(z)), \qquad E(z) = \gamma \sum_{j=1}^{M} \left[k(z, a_j^+) - k(z, a_j^-) \right], \tag{9}$$

where $a_j^+ \in \mathcal{A}^+$ and $a_j^- \in \mathcal{A}^-$ index equal-size positive and negative samples, k is a feature-space kernel, and $\gamma > 0$ sets the dipole strength.

Differentiating Eq. 9 gives

$$\nabla_z \log R_{\text{dipole}}(z) = \gamma \sum_{j=1}^{M} \left[k(z, a_j^+) \nabla_z \log k(z, a_j^+) - k(z, a_j^-) \nabla_z \log k(z, a_j^-) \right]. \tag{10}$$

The induced field is kernel-weighted and local: points near positive samples are strongly attracted, points near negative samples are strongly repelled, and points far from the observed samples receive negligible update since when $E(z) \to 0$ we have $R_{\text{dipole}}(z) \to 1$. This locality is important for one-step finetuning because it lets the reward signal guide nearby samples while reducing extrapolated reward gradients far from the evidence in the current batch.

4.2 Drifting Preference Optimization

For a differentiable, non-negative reward $R(z)$, let $x = g_\theta(\epsilon, c)$ and $z = \phi(x)$. The pathwise gradient of the reference-regularized soft RL objective is

$$\nabla_\theta J = \mathbb{E}_\epsilon \left[\left(\nabla_z \log R(z) + \lambda \left(\nabla_z \log p_{\text{base}}(z) - \nabla_z \log p_\theta(z) \right) \right) \nabla_\theta z \right], \tag{11}$$

where $\nabla_\theta z = \nabla_x \phi(x) \nabla_\theta g_\theta(\epsilon, c)$.

Given an on-policy candidate batch $\mathcal{B} = \{x_i\}_{i=1}^{K}$ from $x_i = g_\theta(\epsilon_i, c)$, $\epsilon_i \sim \mathcal{N}(0, I)$, we evaluate the target reward and sample M reward-ordered pairs $\mathcal{D}_\theta = \{(x_j^+, x_j^-)\}_{j=1}^{M}$ with $R(z_j^+) \geq R(z_j^-)$ and $z_j = \phi(x_j)$. The positive and negative samples are

$$\mathcal{A}^+ = \{\phi(x_j^+)\}_{j=1}^{M}, \qquad \mathcal{A}^- = \{\phi(x_j^-)\}_{j=1}^{M}. \tag{12}$$

The reward-score term in Eq. 11 is replaced by the dipole preference field (Eq. 9):

$$\nabla_z \log R(z) \approx V_{\text{pref}}(z) = \nabla_z \log R_{\text{dipole}}(z; \mathcal{A}^+, \mathcal{A}^-). \tag{13}$$

By treating samples for $\nabla_z \log p_{\text{base}}$ and $\nabla_z \log p_\theta(z)$ as positive and negative terms, respectively, we use the mean-shift estimate in Eq. 5 and write (with $\mathcal{R}$ and $\mathcal{Z}$ being independent sample sets from the two terms):

$$V_{\text{ref}}(z_i) = \widehat{\mu}_{\mathcal{R}}(z_i) - \widehat{\mu}_{\mathcal{Z}}(z_i). \tag{14}$$

The final DrPO field is

$$V_{\text{DrPO}}(z_i) = V_{\text{pref}}(z_i) + \lambda V_{\text{ref}}(z_i). \tag{15}$$

For $z_i = \phi(g_\theta(\epsilon_i, c))$, the detached target is

$$z_i^\star = \text{sg}\left(z_i + \eta V_{\text{DrPO}}(z_i)\right), \tag{16}$$

with loss

$$\mathcal{L}_{\text{DrPO}}(\theta) = \frac{1}{2K} \sum_{i=1}^{K} \|z_i - z_i^\star\|_2^2. \tag{17}$$

The reward model, reference generator, and feature extractor are used only during training; the deployed model remains the same one-step generator.

Non-differentiable rewards. DrPO only requires the target reward R to rank generated samples. Therefore, the same construction applies when R is non-differentiable, such as rule-based evaluators or task-specific correctness checks: we use R to form $\mathcal{A}^+$ and $\mathcal{A}^-$, and the subsequent dipole drift is computed entirely in feature space.

Offline variant. We also make a naive attempt to extend DrPO to offline preference finetuning. Given offline preference pairs, we replace the online feature sets $\mathcal{A}^+$ and $\mathcal{A}^-$ with chosen and rejected features from the dataset; the reference term and detached feature regression remain unchanged. This removes the online reward-ranking step while keeping the same drifting update.

5 Experiments

5.1 Experimental Setup

Base models. We use SD-Turbo [Sauer et al., 2024] (*license: Stability AI Community License*) as the main one-step text-to-image base model, and evaluate SDXL-Turbo [Podell et al., 2024, Sauer et al., 2024] (*license: Stability AI Community License*) in the base-model ablation. The two models use different architectures and latent spaces, so the SDXL-Turbo experiment tests whether the same dipole-drift update transfers beyond the SD-Turbo setting.

Target reward models. The default target reward is PickScore [Kirstain et al., 2023] (*license: MIT*). We also use HPSv3 [Ma et al., 2025] (*license: MIT*) to test training with a large VLM-based reward. In the target-reward ablation, we replace or combine PickScore with Aesthetic Score (AES) [Schuhmann, 2022] (*license: MIT*), HPSv2 [Wu et al., 2023] (*license: Apache-2.0*), ImageReward (IR) [Xu et al., 2023] (*license: Apache-2.0*), and CLIP [Radford et al., 2021] (*license: MIT*). AES is unconditional, while the other rewards are text-conditioned.

Prompts and evaluation sets. Online experiments use prompts from the Pick-a-Pic v2 training split [Kirstain et al., 2023]. The primary evaluation uses the Pick-a-Pic v2 test split with 424 prompts and five matched seeds. We also evaluate on Parti-Prompts [Yu et al., 2022], which contains 1,632 prompts. For non-differentiable reward experiments, we use 553 GenEval prompts covering object presence, counting, color, position, and attribute binding [Ghosh et al., 2023].

Method	Inference Steps	Reward Grad	Pick-a-Pic v2 Test			Parti-Prompts		
			PS ↑	AES ↑	IR ↑	PS ↑	AES ↑	IR ↑
SDXL base	50	–	22.15	6.104	6.85	22.64	5.761	7.24
SDXL-DPO	50	–	22.57	6.076	9.38	22.95	5.811	10.66
SDXL-Turbo (base)	1	–	22.45	6.059	9.36	22.77	5.693	9.13
DRaFT	1	Yes	**24.45**	6.712	**12.70**	**24.34**	6.485	**12.66**
VGG-Flow$_{\text{1-step}}$	1	Yes	24.27	6.490	12.19	23.98	6.200	11.40
DPO$_{\text{1-step}}$	1	No	22.77	6.227	10.44	22.85	6.019	11.25
PSO	1	No	22.56	6.092	8.97	22.87	5.744	9.17
GRPO$_{\text{1-step}}$	1	No	22.50	6.077	9.57	22.80	5.710	9.27
DrPO	1	No	23.66	**6.717**	12.46	23.71	**6.665**	12.60

Table 1: **Results on SDXL-Turbo.** We report test-time generator calls as inference steps and indicate whether each method backpropagates through the target reward during training. For multi-step SDXL baselines, 50×2 denotes classifier-free guidance with two model evaluations per denoising step. PS denotes PickScore $\times 100$ and IR denotes ImageReward $\times 10$.

Baselines. We compare with two groups of finetuning methods: (1) *reward-gradient-based methods*, including DRaFT and VGG-Flow$_{\text{1-step}}$ [Clark et al., 2024, Liu et al., 2025]; and (2) *reward-gradient-free methods*, including DPO$_{\text{1-step}}$, PSO, and GRPO$_{\text{1-step}}$ [Rafailov et al., 2023, Wallace et al., 2024, Shao et al., 2024]. For methods not originally designed for deterministic one-step generators, we adapt their core objectives to the same finetuning setting and evaluate with matched prompts and seeds. Details of the one-step adaptations are given in the appendix.

Metrics. We evaluate text-to-image generation with scalar preference and quality metrics, including PickScore, AES, HPSv2, HPSv3, and ImageReward. FID is reported when matched references are available [Heusel et al., 2017]; GenEval uses the standard compositional scores [Ghosh et al., 2023]. We also report pairwise win rates using Qwen3-VL [Qwen Team, 2025] as an external VLM judge, following recent VQA/MLLM-based evaluation protocols [Hu et al., 2023, Lin et al., 2024, Ku et al., 2024, Lee et al., 2024]. Qwen3-VL compares image pairs generated from the same prompt and outputs `A`, `B`, or `tie`; we randomize order to reduce position bias [Zheng et al., 2023, Shi et al., 2025] and aggregate parsed JSON decisions into win/loss/tie rates. Details are provided in Appendix B.4.

Feature extractors. Our default feature extractor is the 340M-parameter latent-MAE encoder from Drifting Models [Deng et al., 2026]. We ablate this choice against raw latent features and DINOv2 features extracted from VAE-decoded images. We also compare multi-layer latent-MAE variants: latent-MAE$_1$ is the default multi-feature variant, latent-MAE$_2$ uses the last three layers, and latent-MAE$_3$ uses only the final layer. For SDXL-Turbo, where the SD-VAE-specific latent-MAE is not directly applicable, we use decoded image-space MAE features [He et al., 2022].

Implementation details. We finetune LoRA parameters [Hu et al., 2022] with rank 16. Unless otherwise specified, training uses a per-GPU mini-batch size of 24 on 4 GPUs with 8 gradient-accumulation steps, giving an effective batch size of 768. Each DrPO mini-batch uses one prompt to generate 24 candidates; the reward model samples 12 reward-ordered pairs to construct $\mathcal{A}^+$ and $\mathcal{A}^-$. We use AdamW with learning rate 1×10^{-4}. For HPSv3 experiments, we use learning rate 3×10^{-5} and reduce the number of candidate images per prompt to 16. For the DRaFT+HPSv3 baseline, we use mini-batch size 2 with 64 gradient-accumulation steps to avoid out-of-memory errors while keeping the same effective batch size.

5.2 Results

Main results. Table 1 evaluates SDXL-Turbo, a SDXL-family one-step backbone. DrPO improves PickScore, AES, and ImageReward over the base model on both Pick-a-Pic v2 and Parti-Prompts while using the target reward only for ranking. Direct reward-gradient methods such as DRaFT and VGG-Flow$_{\text{1-step}}$ still lead several scalar-reward columns, but require backpropagation through the target reward. Among reward-gradient-free one-step preference methods, DrPO gives the largest gains. Table 2 shows that DrPO can be generalized to different base models. Qualitative results are shown in Fig. 5, Fig. 7, and Fig. A4.

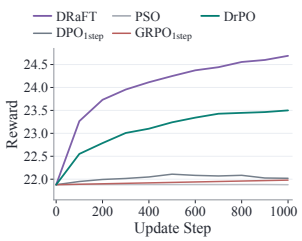


Figure 4: Reward curves on SD-Turbo.

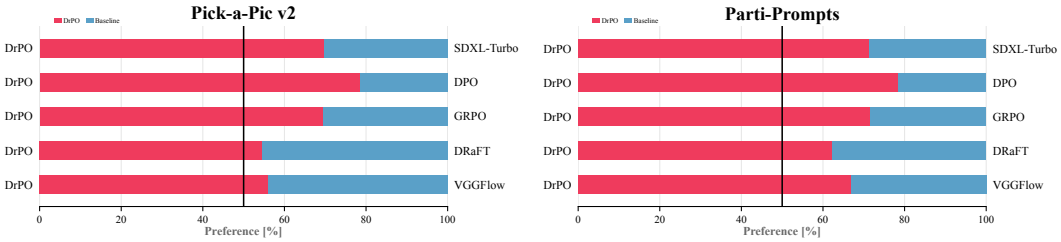


Figure 3: Qwen3-VL pairwise preference evaluation. For each prompt, Qwen3-VL compares two matched generations under the same prompt and selects the cleaner, more coherent prompt realization; ties are allowed and A/B order is randomized. Red indicates DrPO preference, blue indicates the compared baseline preference, and the vertical line marks 50%. The full judge prompt is provided in Appendix B.4.

Method	Inference Steps	Reward Grad	Pick-a-Pic v2 Test			Parti-Prompts		
			PS ↑	AES ↑	IR ↑	PS ↑	AES ↑	IR ↑
SD1.5	50	–	20.79	5.455	1.22	21.49	5.358	2.25
LCM-SD1.5	4	–	20.50	5.497	-3.08	21.15	5.396	-1.94
SD2.1	50	–	21.09	5.645	2.49	21.77	5.547	3.97
SD-Turbo (base)	1	–	21.88	6.054	5.75	22.29	5.758	5.37
DRaFT	1	Yes	**24.69**	**6.820**	9.63	**23.07**	**6.516**	7.72
VGG-Flow$_{\text{1-step}}$	1	Yes	23.73	6.378	7.74	22.99	6.027	6.50
DPO$_{\text{1-step}}$	1	No	22.02	6.080	5.95	22.39	5.793	5.21
PSO	1	No	21.88	6.059	5.80	22.29	5.763	5.42
GRPO$_{\text{1-step}}$	1	No	21.98	6.077	6.08	22.35	5.779	5.65
DrPO	1	No	23.49	6.485	9.54	22.99	6.284	7.46

Table 2: **Results on SD-Turbo.** We report test-time generator calls as inference steps and indicate whether each method backpropagates through the target reward during training. PS denotes PickScore $\times 100$ and IR denotes ImageReward $\times 10$.

Qwen3-VL pairwise preference. Figure 3 reports pairwise judgments from Qwen3-VL for matched generations under the same prompt. Unlike scalar reward metrics, this evaluation asks a VLM judge to compare two complete images with respect to prompt fidelity, visual coherence, artifacts, and aesthetics. The judge outputs `A`, `B`, or `tie`; A/B order is randomized, and decisions are aggregated into win/loss/tie rates. The red bars show that DrPO is consistently preferred over one-step baselines on both Pick-a-Pic v2 and Parti-Prompts, indicating better prompt realization under an external VLM evaluator. The judge prompt is provided in Appendix B.4.

Efficiency with large reward models. Figure 6 profiles training with HPSv3. Under the matched effective-batch setting, DrPO removes HPSv3 reward-model backpropagation and reduces update time from 21.62s to 6.17s, giving a 3.51× speedup.

Metric	DRaFT	**DrPO**
Reward grad	Yes	No
Effective batch	192	192
Update time (s) ↓	21.62	**6.17**
Speedup ↑	1.00×	**3.51**×
Backward (s)	9.99	**0.34**

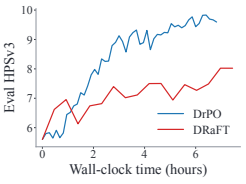


Figure 6: Efficiency comparison on HPSv3.

Non-differentiable rewards. Figure 9 evaluates DrPO with GenEval scores as the target reward. GenEval provides non-differentiable task-level scores for compositional constraints such as counting, color, position, and attribute binding [Ghosh et al., 2023]. DrPO uses these scores to rank candidates and form $\mathcal{A}^+$ and $\mathcal{A}^-$, with no change to the feature-space drift update.

Ablation on target rewards. Figure 8 and Table A6 show DrPO results with different target reward models. The drift estimator and optimization objective are kept fixed, suggesting that DrPO is robust to the choice of reward model.

Ablation on drift design. Table 4 ablates candidate count, feature extractor, kernel choice, and velocity scale. Increasing the candidate count improves PickScore and AES, suggesting that a larger on-policy pool gives more stable pairwise estimates. Latent-MAE features outperform raw latent features, while raw latents substantially degrade performance; the drift benefits from a representation whose local neighborhoods track semantic image changes. DrPO is not highly sensitive to the kernel choice in this range. For the velocity scale η, larger values do not necessarily improve performance.

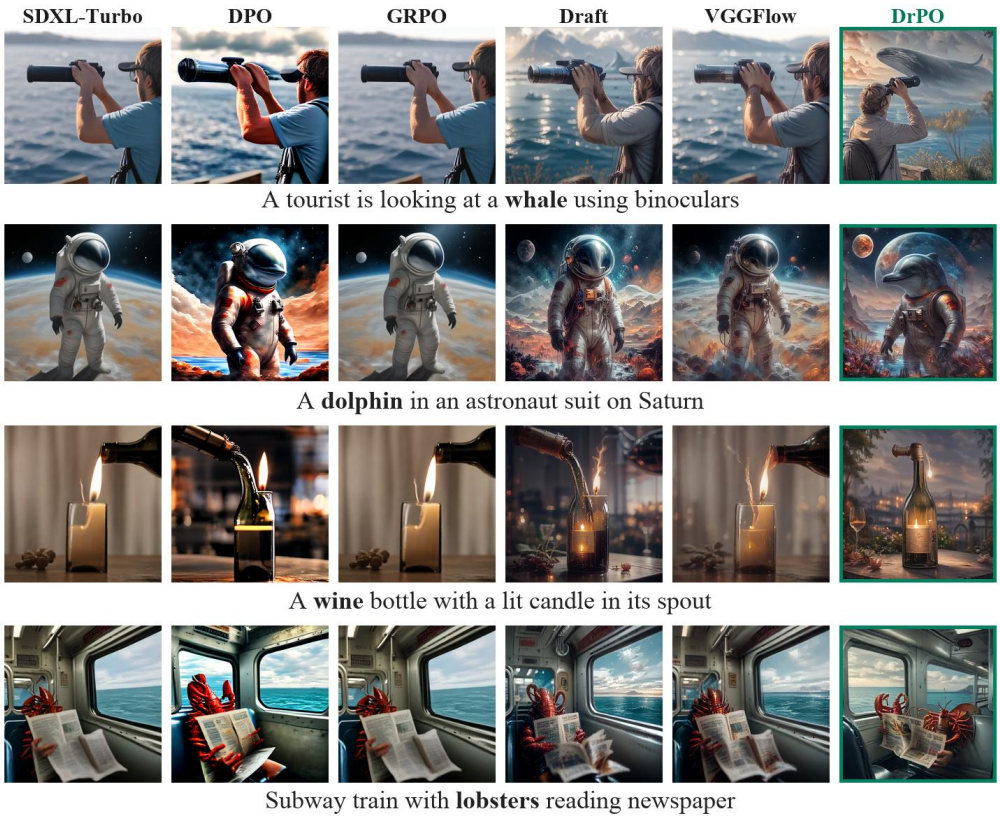


Figure 5: Qualitative comparison on Pick-a-Pic v2 prompts for SDXL-Turbo. Rows compare the SDXL-Turbo base, one-step preference baselines, and DrPO variants under matched prompts and seeds. The examples complement the SDXL-Turbo quantitative comparison in Table 1.

Candidates	PS ↑	AES ↑
SD-Turbo	21.88	6.054
$K = 16$	23.24	6.409
$K = 24$	23.53	6.552
$K = 32$	**23.57**	**6.599**

(a) Batch generation

Feature	PS ↑	AES ↑
SD-Turbo	21.88	6.054
latent-MAE$_1$	**23.55**	6.513
latent-MAE$_2$	23.50	**6.526**
latent-MAE$_3$	23.48	6.506
Latent	20.52	4.543
VAE-dec.+DINOv2	22.28	6.252

(b) Feature map

Kernel	PS ↑	AES ↑
SD-Turbo	21.88	6.054
Cosine	**23.63**	6.509
RBF	23.50	6.590
Exponential	23.53	**6.594**
Laplacian	23.51	6.515

(c) Kernel

Weight	PS ↑	AES ↑
SD-Turbo	21.88	6.054
$\beta = 1000$	23.51	6.510
$\beta = 3000$	**23.53**	**6.542**
$\beta = 5000$	23.51	6.485
$\beta = 10000$	23.46	6.444

(d) Velocity scale η

Table 4: Ablations on different design choices.

Ablation on reference regularization. Table 5 compares the proposed reference drift with a perceptual regularization baseline. Perceptual loss anchors each generated sample to its paired reference image, while reference drift uses local neighborhoods from the base and current generators in the same feature geometry. Reference drift gives small but consistent gains over the perceptual baseline on most alignment metrics.

Offline preference setting. Figure 11 compares a naive offline extension of DrPO with offline DPO$_{\text{1-step}}$ variants using the same SD-Turbo base and Pick-a-Pic v2 test protocol. Here $\mathcal{A}^+$ and $\mathcal{A}^-$ come from fixed preference pairs rather than online reward ranking. Offline DrPO improves over the base model, while the adapted DPO$_{\text{1-step}}$ variants remain closer to SD-Turbo. This separates the effect of the feature-drift update from online candidate selection.

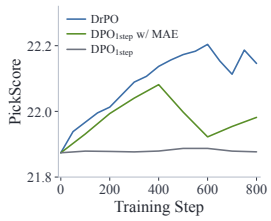


Figure 11: Offline DrPO Convergence.

6 Discussions

Assumption on the dipole reward model. The dipole reward model is compatible with the Bradley–Terry view used in RLHF, where pairwise preferences are induced by scalar reward differences.

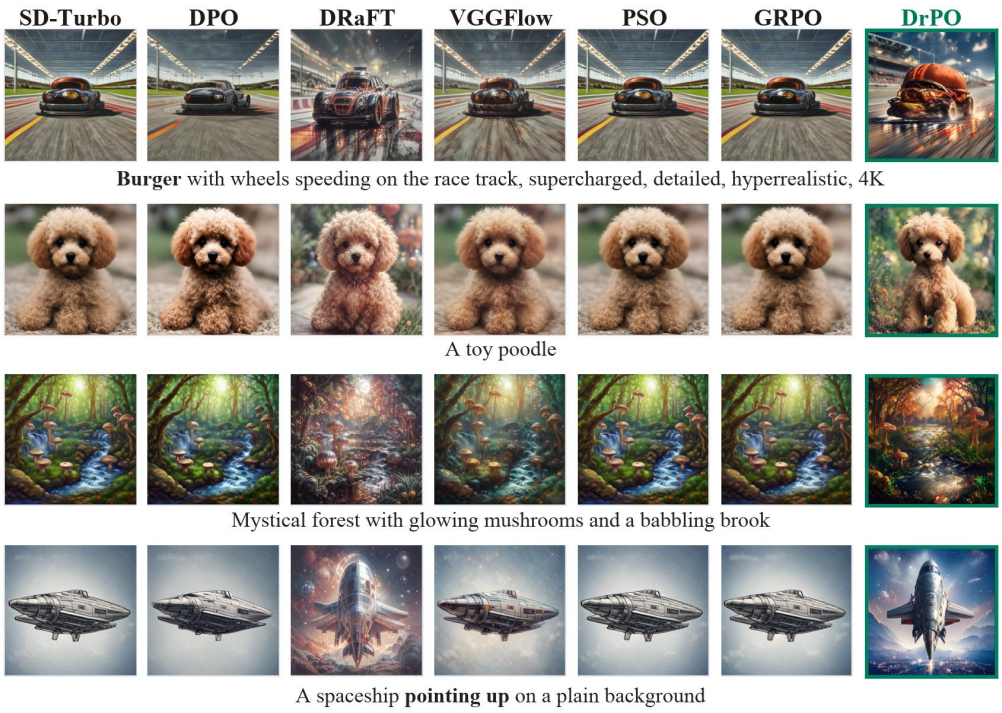


Figure 7: Qualitative headline comparison on selected Pick-a-Pic v2 prompts for SD-Turbo. The grid follows the same row-caption layout as the SDXL-Turbo qualitative comparison. DrPO improves preference-aligned visual quality over the SD-Turbo base while preserving one-step inference.

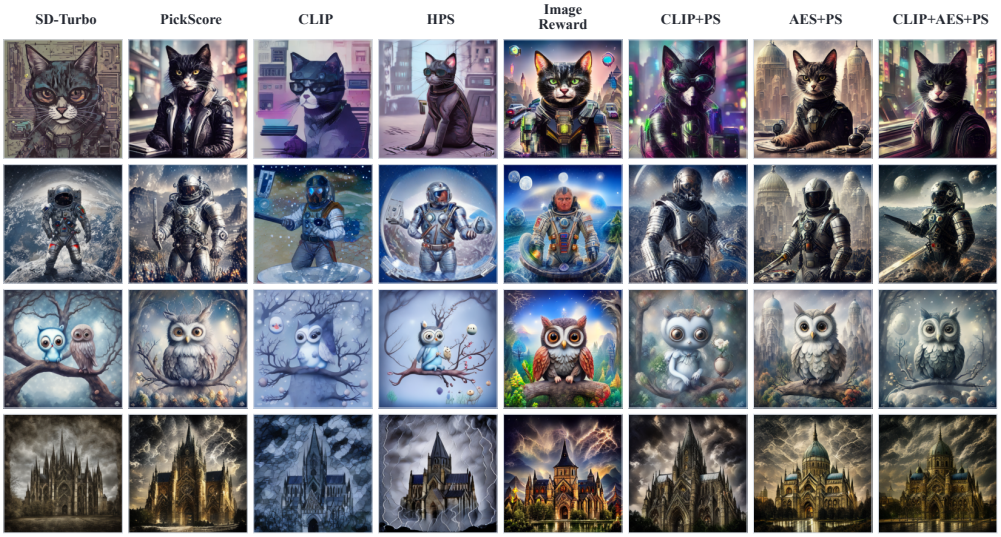


Figure 8: Qualitative comparison across reward models. Each row generated with identical prompt; metrics in Table A6.

In DrPO, this scalar reward is constructed non-parametrically from the current preference pairs: preferred samples define attraction and rejected samples define repulsion under the same feature-space kernel. This makes the feature map important. If ϕ does not encode the attribute being optimized, such as counting, layout, typography, or fine-grained identity, the resulting drift may be smooth in feature space but weakly aligned with the target preference.

Challenges in offline finetuning. Our offline experiment is a naive extension of the online update. In the online setting, preference pairs are sampled from the current generator, so the dipole field is built near the distribution being updated. In the offline setting, the chosen and rejected samples can be

Metric	SD-Turbo	**+DrPO**
Single ↑	98.8	**100.0**
Two ↑	46.5	**55.6**
Count ↑	33.8	**42.5**
Colors ↑	83.8	**87.2**
Position ↑	8.0	**13.0**
Color Attr. ↑	9.0	**13.0**

Table 3: Performance of DrPO on GenEval tasks.

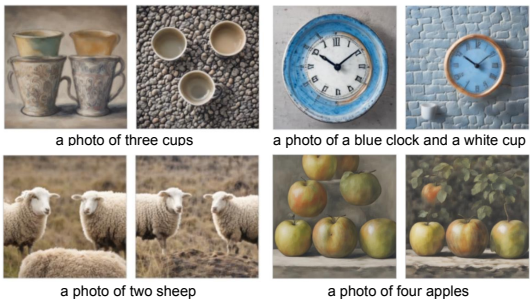


Figure 9: Qualitative examples from online DrPO on GenEval prompts.

Objective	PickScore ↑	CLIP ↑	AES ↑	HPSv2 ↑
No reference	23.55	25.88	6.603	34.85
Perceptual loss	23.42	25.98	6.455	34.90
Ref. drift loss (ours)	**23.49**	**26.22**	**6.485**	**35.07**

Table 5: Reference-objective comparison on Pick-a-Pic v2 test prompts.

Figure 10: Effect of the reference term during finetuning.

off-policy and sparse, and many preference datasets contain only one pair per prompt. This makes the local non-parametric field more fragile. Better offline variants may need larger pair neighborhoods, prompt-conditioned retrieval, or an aggregated field over multiple preference pairs.

Limitations and future work. DrPO inherits the limitations of the target reward used for ranking. If a reward misses prompt attributes or encodes undesirable aesthetic tradeoffs, the resulting preference field can steer the generator in that direction. Training also still requires candidate generation, reward scoring, and feature extraction, so the computational advantage is largest when reward-model backpropagation is expensive or unavailable. The current implementation uses LoRA finetuning and finite candidate batches; scaling candidate selection, feature maps, and reward mixtures are useful directions for future work.

7 Concluding Remarks

We presented DrPO, a preference-finetuning method for one-step text-to-image generators that avoids reward backpropagation. DrPO uses the target reward to rank on-policy samples, forms reward-ordered preference pairs, and converts them into a dipole drift in feature space. A reference drift from the frozen base generator stabilizes the update, and a detached feature regression trains the current generator along the resulting field. The method does not require policy likelihoods, denoising trajectories, or differentiating through the target reward, and inference remains a single generator call. Across one-step generators and reward signals, DrPO improves reward-gradient-free preference finetuning, works with large and non-differentiable rewards, and gives an initial path toward offline preference tuning.

References

Yuntao Bai, Saurav Kadavath, Sandipan Kundu, Amanda Askell, Jackson Kernion, Andy Jones, Anna Chen, Anna Goldie, Azalia Mirhoseini, Cameron McKinnon, et al. Constitutional AI: Harmlessness from AI feedback. *arXiv preprint arXiv:2212.08073*, 2022. 3

Kevin Clark, Paul Vicol, Kevin Swersky, and David J. Fleet. Directly fine-tuning diffusion models on differentiable rewards. In *ICLR*, 2024. 3, 6, 16

Florinel-Alin Croitoru, Vlad Hondru, Radu Tudor Ionescu, Nicu Sebe, and Mubarak Shah. Curriculum direct preference optimization for diffusion and consistency models. In *CVPR*, 2025. 2

Mingyang Deng, He Li, Tianhong Li, Yilun Du, and Kaiming He. Generative modeling via drifting. *arXiv preprint arXiv:2602.04770*, 2026. 1, 2, 3, 6, 13

Luca Eyring, Shyamgopal Karthik, Karsten Roth, Alexey Dosovitskiy, and Zeynep Akata. ReNO: Enhancing one-step text-to-image models through reward-based noise optimization. In *NeurIPS*, 2024. 3

Zhengyang Geng, Mingyang Deng, Xingjian Bai, J Zico Kolter, and Kaiming He. Mean flows for one-step generative modeling. *arXiv preprint arXiv:2505.13447*, 2025. 1

Dhruba Ghosh, Hannaneh Hajishirzi, and Ludwig Schmidt. GenEval: An object-focused framework for evaluating text-to-image alignment. In *NeurIPS*, 2023. 5, 6, 7

Tuomas Haarnoja, Aurick Zhou, Pieter Abbeel, and Sergey Levine. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. In *ICML*, 2018. 4

Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *CVPR*, 2022. 6

Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. GANs trained by a two time-scale update rule converge to a local Nash equilibrium. In *NeurIPS*, 2017. 6

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 *ICLR*, 2022. 6

Yushi Hu, Benlin Liu, Jungo Kasai, Yizhong Wang, Mari Ostendorf, Ranjay Krishna, and Noah A. Smith. Tifa: Accurate and interpretable text-to-image faithfulness evaluation with question answering. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 20406–20417, October 2023. 6

Zhou Jiang, Yandong Wen, and Zhen Liu. Rethinking preference alignment for diffusion models with classifier-free guidance. *arXiv preprint arXiv:2602.18799*, 2026. 3

Jeeyung Kim, Ze Wang, and Qiang Qiu. Model-agnostic human preference inversion in diffusion models. *arXiv preprint arXiv:2404.00879*, 2024. 3

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 *NeurIPS*, 2023. 5

Max Ku, Dongfu Jiang, Cong Wei, Xiang Yue, and Wenhu Chen. VIEScore: Towards explainable metrics for conditional image synthesis evaluation. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 12268–12290, Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-long.663. 6

Seongyun Lee, Seungone Kim, Sue Park, Geewook Kim, and Minjoon Seo. Prometheus-vision: Vision-language model as a judge for fine-grained evaluation. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 11286–11315, Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.findings-acl.672. 6

Zhiqiu Lin, Deepak Pathak, Baiqi Li, Jiayao Li, Xide Xia, Graham Neubig, Pengchuan Zhang, and Deva Ramanan. Evaluating text-to-visual generation with image-to-text generation. *arXiv preprint arXiv:2404.01291*, 2024. 6

Zhen Liu, Tim Z Xiao, Carles Domingo-Enrich, Weiyang Liu, and Dinghuai Zhang. Value gradient guidance for flow matching alignment. *arXiv preprint arXiv:2512.05116*, 2025. 6, 16

Cheng Lu and Yang Song. Simplifying, stabilizing and scaling continuous-time consistency models. *arXiv preprint arXiv:2410.11081*, 2024. 2

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. 1

Weijian Luo. Diff-Instruct++: Training one-step text-to-image generator model to align with human preferences. *Transactions on Machine Learning Research*, 2025. 2

Weijian Luo, Colin Zhang, Debing Zhang, and Zhengyang Geng. David and goliath: Small one-step model beats large diffusion with score post-training. In *ICML*, 2025a. 2

Yihong Luo, Tianyang Hu, Weijian Luo, Kenji Kawaguchi, and Jing Tang. Reward-instruct: A reward-centric approach to fast photo-realistic image generation. In *NeurIPS*, 2025b. 2

Yihong Luo, Tianyang Hu, Yifan Song, Jiacheng Sun, Zhenguo Li, and Jing Tang. Adding additional control to one-step diffusion with joint distribution matching. In *ICCV*, 2025c. 2

Yuhang Ma, Xiaoshi Wu, Keqiang Sun, and Hongsheng Li. HPSv3: Towards wide-spectrum human preference score. In *ICCV*, 2025. 5

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 *ICLR*, 2025. 2, 16

Thuan Hoang Nguyen and Anh Tran. Swiftbrush: One-step text-to-image diffusion model with variational score distillation. In *CVPR*, 2024. 2

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. In *NeurIPS*, 2022. 3

Dustin Podell, Zion English, Kyle Lacey, Andreas Blattmann, Tim Dockhorn, Jonas Müller, Joe Penna, and Robin Rombach. SDXL: Improving latent diffusion models for high-resolution image synthesis. In *ICLR*, 2024. 5

Qwen Team. Qwen3-VL technical report. *arXiv preprint arXiv:2511.21631*, 2025. 6, 15

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *ICML*, 2021. 5

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D. Manning, Stefano Ermon, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. In *NeurIPS*, 2023. 3, 6

Axel Sauer, Tero Karras, Samuli Laine, Andreas Geiger, and Timo Aila. Stylegan-t: Unlocking the power of gans for fast large-scale text-to-image synthesis. In *ICML*, 2023. 1

Axel Sauer, Dominik Lorenz, Andreas Blattmann, and Robin Rombach. Adversarial diffusion distillation. In *ECCV*, 2024. 1, 5

Christoph Schuhmann. LAION-Aesthetics. https://laion.ai/blog/laion-aesthetics/, 2022. 5

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017. 3

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. DeepSeekMath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024. 3, 6

Lin Shi, Chiyu Ma, Wenhua Liang, Xingjian Diao, Weicheng Ma, and Soroush Vosoughi. Judging the judges: A systematic study of position bias in LLM-as-a-judge. In *Proceedings of the 14th International Joint Conference on Natural Language Processing and the 4th Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics*, pages 292–314, Mumbai, India, December 2025. The Asian Federation of Natural Language Processing and The Association for Computational Linguistics. doi: 10.18653/v1/2025.ijcnlp-long.18. 6

Yang Song, Prafulla Dhariwal, Mark Chen, and Ilya Sutskever. Consistency models. In *ICML*, 2023. 1

Nisan Stiennon, Long Ouyang, Jeffrey Wu, Daniel M. Ziegler, Ryan Lowe, Chelsea Voss, Alec Radford, Dario Amodei, and Paul F. Christiano. Learning to summarize with human feedback. In *NeurIPS*, 2020. 3

Bram Wallace, Meihua Dang, Rafael Rafailov, Linqi Zhou, Aaron Lou, Senthil Purushwalkam, Stefano Ermon, Caiming Xiong, Shafiq Joty, and Nikhil Naik. Diffusion model alignment using direct preference optimization. In *CVPR*, 2024. 3, 6

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. 5

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 *NeurIPS*, 2023. 5

Tianwei Yin, Michaël Gharbi, Taesung Park, Richard Zhang, Eli Shechtman, Fredo Durand, and William T Freeman. Improved distribution matching distillation for fast image synthesis. In *NeurIPS*, 2024a. 2

Tianwei Yin, Michaël Gharbi, Richard Zhang, Eli Shechtman, Fredo Durand, William T Freeman, and Taesung Park. One-step diffusion with distribution matching distillation. In *CVPR*, 2024b. 1, 2

Jiahui Yu, Yuanzhong Xu, Jing Yu Koh, Thang Luong, Gunjan Baid, Zirui Wang, Vijay Vasudevan, Alexander Ku, Yinfei Yang, Burcu Karagol Ayan, et al. Scaling autoregressive models for content-rich text-to-image generation. *Transactions on Machine Learning Research*, 2022. 5

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric P. Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. Judging llm-as-a-judge with mt-bench and chatbot arena. In *Advances in Neural Information Processing Systems*, volume 36, 2023. 6

Appendix

Table of Contents

A Finite-Batch Drift Estimator **13**

B Experimental Details **15**
- B.1 Target Reward Models 15
- B.2 Data and Evaluation Sets 15
- B.3 Sampling and Reproducibility 15
- B.4 Qwen3-VL Pairwise Evaluation Prompt 15

C Baselines and Ablations **16**
- C.1 DRaFT Reward-Gradient Baseline 16
- C.2 VGG-Flow$_{\text{1-step}}$ Reward-Gradient Target Baseline 16
- C.3 DPO$_{\text{1-step}}$ Pairwise Baselines 17
- C.4 GRPO$_{\text{1-step}}$ Policy-Gradient Baseline 17

D Additional Qualitative Results **19**

A Finite-Batch Drift Estimator

Algorithm A1 DrPO with stabilized finite-batch drift estimator

Require: Generator g_θ, frozen reference model g_{ref}, feature map ϕ, reward model R, drift radii Ω, reference weight λ

for each optimization step **do**
  Sample condition c and noises $\{\epsilon_i\}_{i=1}^K$
  Generate current samples $x_i = g_\theta(\epsilon_i, c)$ and reference samples $x_i^{\text{ref}} = g_{\text{ref}}(\epsilon_i, c)$
  Compute features $Z = \{\phi(x_i)\}_{i=1}^K$ and $\mathcal{R} = \{\phi(x_i^{\text{ref}})\}_{i=1}^K$
  Rank $\{x_i\}_{i=1}^K$ with R and form preference pairs $\mathcal{D}_\theta = \{(x_j^+, x_j^-)\}_{j=1}^M$
  Set $\mathcal{A}^+ = \{\phi(x_j^+)\}_{j=1}^M$ and $\mathcal{A}^- = \{\phi(x_j^-)\}_{j=1}^M$
  $V_{\text{pref}} \leftarrow \textsc{DriftField}(Z, \mathcal{A}^+, \mathcal{A}^-, \Omega)$
  $V_{\text{ref}} \leftarrow \textsc{DriftField}(Z, \mathcal{R}, \text{sg}(Z), \Omega)$
  $V_{\text{DrPO}} \leftarrow V_{\text{pref}} + \lambda V_{\text{ref}}$
  $Z^\star \leftarrow \text{sg}(Z + \eta V_{\text{DrPO}})$
  Update θ with $\mathcal{L}_{\text{DrPO}} = \frac{1}{K}\sum_i \|z_i - z_i^\star\|_2^2$
end for

function $\textsc{DriftField}(Z, B^+, B^-, \Omega)$
  Normalize Z, B^+, B^- by the batch feature scale in Eq. 21
  $V \leftarrow 0$
  for $\rho \in \Omega$ **do**
    Compute double-softmax affinities $A^{+,\rho}, A^{-,\rho}$ by Eq. 22
    Compute cross-mass field $V^{(\rho)}$ by Eq. 24
    $V \leftarrow V + \text{RMSNorm}(V^{(\rho)})$
  end for
  return V
end function

This section gives the stabilized finite-batch drift estimator used in our experiments. The main text describes the conceptual fields V_{pref} and V_{ref}; in implementation, both are computed with the same mini-batch drifting estimator inherited from Drifting Models [Deng et al., 2026].

Batch construction. Let $X = \{x_i\}_{i=1}^K$ be the current generated features and $\bar{X} = \mathrm{sg}(X)$. For the preference branch, the positive and negative batches are

$$B^+_{\mathrm{pref}} = \mathcal{A}^+, \qquad B^-_{\mathrm{pref}} = \mathcal{A}^-. \tag{18}$$

For the reference branch, they are

$$B^+_{\mathrm{ref}} = \mathcal{R}, \qquad B^-_{\mathrm{ref}} = \bar{X}, \tag{19}$$

where $\mathcal{R}$ contains features from the frozen reference generator. The detached current batch $\bar{X}$ gives the current-model correction in V_{ref}. Matching self-pairs are masked in the reference branch.

Feature normalization. For a query batch $X = \{x_i\}$ and a support batch $B = \{b_j\}$, we first compute distances using detached queries,

$$\Delta_{ij} = \| \mathrm{sg}(x_i) - b_j \|_2. \tag{20}$$

A single feature scale is estimated by

$$s = \max\left(\epsilon_s, \frac{\mathrm{Mean}_{i,j}(w_j \Delta_{ij})}{\mathrm{Mean}_j(w_j)}\right), \tag{21}$$

and features are normalized coordinate-wise by $s/\sqrt{d}$ before constructing the drift target. This normalization makes the estimator less sensitive to the absolute scale of the frozen feature extractor.

Double-softmax affinity. For each radius ρ, let $D_{ij} = \Delta_{ij}/s$. We use the stabilized affinity from Drifting Models,

$$A^{(\rho)}_{ij} = \sqrt{\mathrm{softmax}_j(-D_{ij}/\rho)\,\mathrm{softmax}_i(-D_{ij}/\rho)}\, w_j. \tag{22}$$

The row softmax normalizes supports for each query, while the column softmax normalizes queries for each support; their geometric mean gives the "double-softmax" affinity used in all drift branches.

Cross-mass drift. After partitioning the supports into positive and negative batches, the radius-specific field is computed with the cross-mass weighting

$$W^{+,\rho}_{ij} = \left(\sum_{\ell \in B^-} A^{-,\rho}_{i\ell}\right) A^{+,\rho}_{ij}, \qquad W^{-,\rho}_{ij} = \left(\sum_{\ell \in B^+} A^{+,\rho}_{i\ell}\right) A^{-,\rho}_{ij}, \tag{23}$$

$$V^{(\rho)}_i = \sum_{j \in B^+} W^{+,\rho}_{ij} b^+_j - \sum_{j \in B^-} W^{-,\rho}_{ij} b^-_j. \tag{24}$$

This is the finite-batch implementation of the attraction–repulsion fields in the main text.

RMS normalization and multi-radius aggregation. Each radius-specific field is normalized by its RMS magnitude over particles and feature coordinates,

$$\widehat{V}^{(\rho)} = \frac{V^{(\rho)}}{\sqrt{\mathrm{Mean}_{i,c}\left((V^{(\rho)}_{i,c})^2\right) + \epsilon_v}}, \tag{25}$$

and the final field is

$$V = \sum_{\rho \in \Omega} \widehat{V}^{(\rho)}. \tag{26}$$

The detached target in Eq. 16 is constructed from this aggregated field. In our experiments, both preference and reference branches use the same estimator, with different choices of positive and negative batches.

Auxiliary anchors. Some runs include weak output-space or feature-space anchors to the frozen reference generator:

$$\mathcal{L}_{\mathrm{impl}} = \mathcal{L}_{\mathrm{pref}} + \lambda_{\mathrm{ref}} \mathcal{L}_{\mathrm{ref}} + \lambda_{\mathrm{out}} \mathcal{L}_{\mathrm{out}} + \lambda_{\mathrm{feat}} \mathcal{L}_{\mathrm{feat}}. \tag{27}$$

These anchors are implementation stabilizers and are ablated separately.

B Experimental Details

B.1 Target Reward Models

DrPO only requires the target reward to rank candidate images for the same prompt. We therefore instantiate the online preference construction with several reward backends, including PickScore, CLIP, AES, HPS-family rewards, ImageReward, and mixtures of these scores. Table A6 shows that the optimized model moves toward the chosen target reward: PickScore training gives the largest PickScore gain, CLIP training gives the largest CLIP gain, HPS training gives the largest HPS gain, and ImageReward training gives the largest ImageReward gain.

Table A6: Target-reward-conditioned online DrPO on Pick-a-Pic v2 test prompts. Entries except AES and ImageReward are multiplied by 100.

Target reward model	PickScore ↑	CLIP ↑	AES ↑	IR ↑	HPSv2 ↑
SD-Turbo	21.88	26.47	6.054	5.75	35.80
PickScore	**23.40**	26.28	6.390	10.09	35.24
CLIP	21.04	**28.62**	5.491	3.31	36.82
HPS	21.13	27.82	5.396	5.60	**39.42**
IR	21.54	23.98	6.314	**16.25**	32.56
CLIP+PickScore	23.01	28.60	6.328	10.08	37.36
AES+PickScore	22.78	23.61	**7.429**	8.51	31.38
CLIP+AES+PickScore	22.94	27.18	7.064	10.70	35.08

B.2 Data and Evaluation Sets

Online training. For online DrPO, each training example contains only a prompt. At each update, the current generator samples a candidate batch for the prompt, the target reward ranks the candidates, and the ordered candidates are used to form preference pairs.

Offline preference data. For the offline variant, each record contains a prompt and one or more chosen/rejected image pairs. The chosen and rejected images replace the online ranked candidates when constructing the preference pairs; the reference drift and training loss are unchanged.

GenEval. For GenEval experiments, prompts are paired with structured compositional constraints such as object category, count, color, position, and attribute binding. The GenEval evaluator returns task-level correctness scores, which are used only to rank candidate images during online training.

Evaluation. The main scalar-metric evaluations use the Pick-a-Pic v2 test prompts. Unless otherwise stated, one-step models are evaluated with deterministic sampling and guidance scale 0.0. We report one image per prompt for PickScore, CLIP, AES, HPSv2, HPSv3, and ImageReward; standard errors are estimated by bootstrapping prompts.

B.3 Sampling and Reproducibility

Evaluation supports three sampling modes: the original SD-Turbo checkpoint, a finetuned U-Net checkpoint, and a LoRA adapter. For structured compositional benchmarks, the same harness reads the benchmark metadata and runs the corresponding evaluator. All paper-facing comparisons fix prompt sets, random seeds, sampling configuration, and the distinction between target rewards used for training and metrics used only for evaluation.

B.4 Qwen3-VL Pairwise Evaluation Prompt

For the pairwise preference results in Figure 3, we use Qwen3-VL [Qwen Team, 2025] as a strict text-to-image judge. Each comparison shows two images generated from the same prompt, and the judge returns a JSON decision with ties allowed. The prompt template is:

```
You are a strict text-to-image judge comparing two generated images.

You will see Image A and Image B. They were generated from the same text prompt.

Text prompt:
"""{prompt}"""

Choose the image that is better as a clean, coherent realization of the prompt.

Rubric, in order:
1. Semantic fidelity: prefer the image where the prompt's main subject is immediately recognizable and
↪  the requested attributes, relations, actions, and scene are correctly expressed.
2. Global coherence: prefer one integrated scene over a collage-like, over-stylized, or visually
↪  confused image. Objects should have plausible shape, scale, pose, and spatial relation.
3. Artifact avoidance: strongly penalize malformed bodies, fused objects, duplicated parts, warped
↪  geometry, messy textures, unreadable required text, and unnatural distortions.
4. Aesthetic quality: after the above, prefer tasteful lighting, color harmony, composition, and detail.
↪  Do not choose an image only because it is sharper, glossier, more saturated, or more dramatic.

Decision rule:
- Major semantic or structural errors should usually lose, even if the image looks polished.
- If both satisfy the text similarly, choose the more coherent and artifact-free image.
- If both are genuinely comparable, answer tie.
- Do not prefer A or B because of position or label.
- Return only valid JSON.

Output schema:
{"winner":"A"|"B"|"tie","confidence":0.0-1.0,"reason":"12 words or fewer"}
```

C Baselines and Ablations

This section describes the one-step preference baselines used in our comparisons. All baselines use the same prompt sets, one-step sampler, and evaluation suite as DrPO.

PSO baseline. PSO is a preference-finetuning method for timestep-distilled diffusion models [Miao et al., 2025]. It optimizes pairwise preferences by increasing the relative margin between preferred images and reference images sampled from the distilled model. We include it as a native pairwise baseline for one-step distilled generators.

C.1 DRaFT Reward-Gradient Baseline

DRaFT directly optimizes a differentiable image reward through the generator and decoder [Clark et al., 2024]. In our one-step setting, the U-Net predicts a clean latent

$$\mu_\theta = T(z, f_\theta(z, t, c)),$$

which is decoded and scored by a differentiable reward model R. We use the same one-step sampler, LoRA parameterization, prompt batches, and evaluation protocol as DrPO. The training objective is

$$\mathcal{L}_{\text{DRaFT}} = -\frac{1}{K}\sum_{k=1}^{K} R(D(\mu_{\theta,k}), p) + \lambda_{\text{ref}}\frac{1}{K}\sum_{k=1}^{K}\|\mu_{\theta,k} - \mu_{\text{ref},k}\|_2^2,$$

where D is the VAE decoder and $\mu_{\text{ref},k}$ is the frozen base-model prediction for the same prompt and noise. This baseline represents the standard reward-backpropagation regime: it can exploit the local gradient of a differentiable reward, but it requires storing and backpropagating through the reward network. For SD-Turbo and SDXL-Turbo, the implementation evaluates generated samples in chunks to keep the effective generated batch size matched to DrPO while controlling memory.

C.2 VGG-Flow$_{\text{1-step}}$ Reward-Gradient Target Baseline

VGG-Flow$_{\text{1-step}}$ adapts value-gradient guidance for flow matching alignment to the one-step latent prediction setting [Liu et al., 2025]. Rather than directly maximizing the scalar reward as in DRaFT, it first differentiates the reward with respect to the generated clean latent and then constructs a target latent around the frozen reference prediction:

$$\mu_k^{\text{target}} = \mu_{\text{ref},k} + \eta(t)\lambda_{\text{r}}\,\text{clip}\big(\nabla_{\mu_{\theta,k}} R(D(\mu_{\theta,k}), p)\big)\,.$$

Algorithm A2 $\mathrm{DPO}_{\text{1-step}}$ baseline

Require: Student U-Net f_θ, frozen reference f_{ref}, offline triples (p, y^+, y^-), text encoder E_{text}, one-step transform T, preference map ψ, inverse temperature β

for each optimization step **do**
  Sample (p, y^+, y^-) and noise latent z; set $c = E_{\mathrm{text}}(p)$
  $\mu_\theta \leftarrow T(z, f_\theta(z, t, c)), \quad \mu_{\mathrm{ref}} \leftarrow T(z, f_{\mathrm{ref}}(z, t, c))$
  Compute $h_\theta = \psi(\mu_\theta)$, $h_{\mathrm{ref}} = \psi(\mu_{\mathrm{ref}})$, $h^+ = \psi(y^+)$, $h^- = \psi(y^-)$
  Define $s(a, b) = -\|a - b\|_2^2$
  $g_\theta \leftarrow s(h_\theta, h^+) - s(h_\theta, h^-)$
  $g_{\mathrm{ref}} \leftarrow s(h_{\mathrm{ref}}, h^+) - s(h_{\mathrm{ref}}, h^-)$
  $\mathcal{L} \leftarrow -\log \sigma(\beta(g_\theta - g_{\mathrm{ref}})) + \lambda_{\mathrm{ref}} \|\mu_\theta - \mu_{\mathrm{ref}}\|_2^2$
  Update θ by descending $\nabla_\theta \mathcal{L}$
end for

The student is trained by latent target matching,

$$\mathcal{L}_{\text{VGG-Flow}_{\text{1-step}}} = \frac{1}{K} \sum_{k=1}^{K} \|\mu_{\theta,k} - \mu_k^{\mathrm{target}}\|_2^2 + \lambda_{\mathrm{ref}} \frac{1}{K} \sum_{k=1}^{K} \|\mu_{\theta,k} - \mu_{\mathrm{ref},k}\|_2^2.$$

Here $\eta(t)$ is fixed in the one-step experiments because the generation timestep is fixed, and the reward gradient is norm-clipped before constructing the target. This implementation preserves the one-step deployment interface but keeps the defining reward-gradient path of VGG-Flow. We omit the multi-step value-network consistency branch from the original flow-matching setting, since SD-Turbo and SDXL-Turbo expose only a single distilled transition in our experiments.

C.3 $\mathrm{DPO}_{\text{1-step}}$ Pairwise Baselines

We adapt DPO to one-step generation by defining the preference margin on the clean latent predicted from a fixed noisy latent and prompt. For text embedding c, timestep $t = 999$, and initial noise z, the student and reference U-Nets produce

$$\mu_\theta = T(z, f_\theta(z, t, c)), \qquad \mu_{\mathrm{ref}} = T(z, f_{\mathrm{ref}}(z, t, c)),$$

where $T(\cdot)$ is the deterministic SD-Turbo one-step conversion. Given chosen and rejected targets y^+ and y^- mapped into a preference space ψ, we define

$$g_\theta = s\big(\psi(\mu_\theta), \psi(y^+)\big) - s\big(\psi(\mu_\theta), \psi(y^-)\big),$$

and define g_{ref} analogously using μ_{ref}. The loss is

$$\mathcal{L}_{\mathrm{DPO}_{\text{1-step}}} = -\log \sigma(\beta[g_\theta - g_{\mathrm{ref}}]) + \lambda_{\mathrm{ref}} \|\mu_\theta - \mu_{\mathrm{ref}}\|_2^2.$$

We use $\beta = 30$ and $\lambda_{\mathrm{ref}} = 0$ in the reported runs. We test two choices of ψ: raw VAE latent space and frozen MAE-latent feature space.

C.4 $\mathrm{GRPO}_{\text{1-step}}$ Policy-Gradient Baseline

We include $\mathrm{GRPO}_{\text{1-step}}$ as a policy-gradient-style comparison for one-step generators. Since a one-step model has no denoising trajectory, the policy action is defined in clean latent space. For a prompt embedding c and a group of initial noises $\{z_k\}_{k=1}^K$,

$$\mu_{\theta,k} = T(z_k, f_\theta(z_k, t, c)), \qquad \mu_{\mathrm{ref},k} = T(z_k, f_{\mathrm{ref}}(z_k, t, c)).$$

The action is a perturbed clean latent,

$$a_k = \mathrm{sg}(\mu_{\theta,k}^{\mathrm{old}}) + \sigma_a u_k, \qquad u_k = \frac{z_k - \mathrm{mean}(z_k)}{\mathrm{std}(z_k) + \epsilon}.$$

After decoding a_k and scoring the image, rewards are normalized within the same prompt group:

$$A_k = \mathrm{clip}\left(\alpha_{\mathrm{adv}} \frac{r_k - \mathrm{mean}(r)}{\mathrm{std}(r) + \epsilon_r}, -A_{\max}, A_{\max}\right).$$

Algorithm A3 $\text{GRPO}_{\text{1-step}}$ baseline

Require: Student U-Net f_θ, frozen reference f_{ref}, reward model R, text encoder E_{text}, decoder D, one-step transform T, group size K

for each prompt **do**
  $c \leftarrow E_{\text{text}}(p)$; sample $\{z_k\}_{k=1}^{K}$
  for $k = 1, \dots, K$ **do**
    $\mu_k^{\text{old}} \leftarrow T(z_k, f_\theta(z_k, t, c))$ without gradient
    $a_k \leftarrow \text{sg}(\mu_k^{\text{old}}) + \sigma_a \ \text{normalize}(z_k)$
    $x_k \leftarrow D(a_k); \quad r_k \leftarrow R(x_k, c)$
  end for
  Normalize rewards within the prompt group to obtain $\{A_k\}$
  for $k = 1, \dots, K$ **do**
    Recompute $\mu_k \leftarrow T(z_k, f_\theta(z_k, t, c))$ and $\mu_k^{\text{ref}} \leftarrow T(z_k, f_{\text{ref}}(z_k, t, c))$
    Compute ρ_k from the Gaussian log-probability ratio of a_k
  end for
  Update θ with the clipped policy-gradient loss and reference latent anchor
end for

The current policy mean is recomputed and assigned a Gaussian-style log probability

$$\ell_\theta(a_k) = -\frac{1}{2} \,\text{mean}\left[\left(\frac{a_k - \mu_{\theta,k}}{\sigma_a}\right)^2\right],$$

where the mean is over latent dimensions. With

$$\rho_k = \exp(\text{clip}(\ell_\theta(a_k) - \ell_{\text{old}}(a_k), -M, M)),$$

we use the clipped policy-gradient objective

$$\mathcal{L}_{\text{policy}} = -\frac{1}{K}\sum_{k=1}^{K} \min\left(\rho_k A_k,\ \text{clip}(\rho_k, 1-\epsilon, 1+\epsilon) A_k\right).$$

The total loss is

$$\mathcal{L} = \mathcal{L}_{\text{policy}} + \lambda_{\text{ref}} \frac{1}{K}\sum_{k=1}^{K} \|\mu_{\theta,k} - \mu_{\text{ref},k}\|_2^2 + \beta_{\text{kl}} \frac{1}{K}\sum_{k=1}^{K} (\rho_k - 1 - \log \rho_k).$$

In the reported setting, $\sigma_a = 0.05$, $\epsilon = 0.2$, $M = 10$, $\alpha_{\text{adv}} = 1$, $A_{\max} = 5$, $\lambda_{\text{ref}} = 0.05$, and $\beta_{\text{kl}} = 0$.

D Additional Qualitative Results

Figures A4 and A3 provide the full qualitative grids corresponding to the compact examples in the main text. All samples use matched seed 42.

Reward-model prompts. The compact reward-model matrix in Figure 8 uses four Pick-a-Pic v2 test prompts: “Cyberpunk cat”; “astronaut in space with a two handed sword in plate armor in front of the earth”; “An mystical owl sitting on a tree branch in a magical Forest, art style of nicoletta ceccoli”; and “Gothic cathedral in a stormy night”. The full matrix in Figure A3 additionally includes “smily french fries” and “Raindrop, macro photograph, colorful, reflections, HD, 4k”.

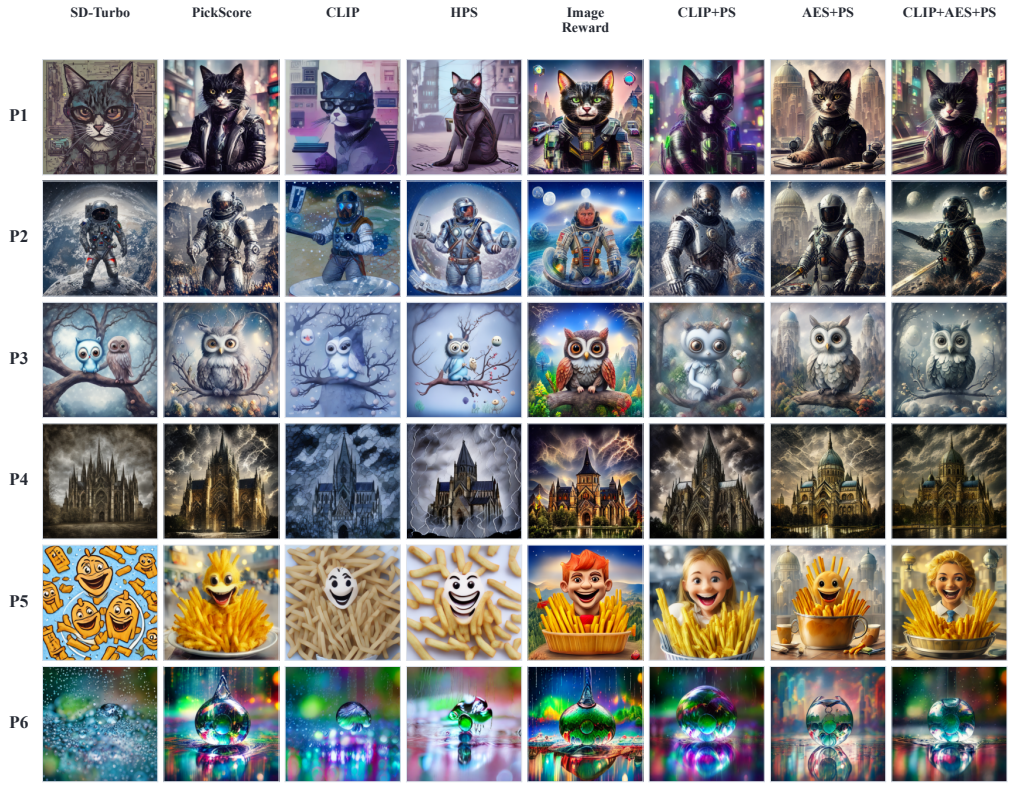


Figure A3: Full reward-model qualitative matrix corresponding to Figure 8. Rows are prompts, columns are target reward models, and all samples use matched seed 42.

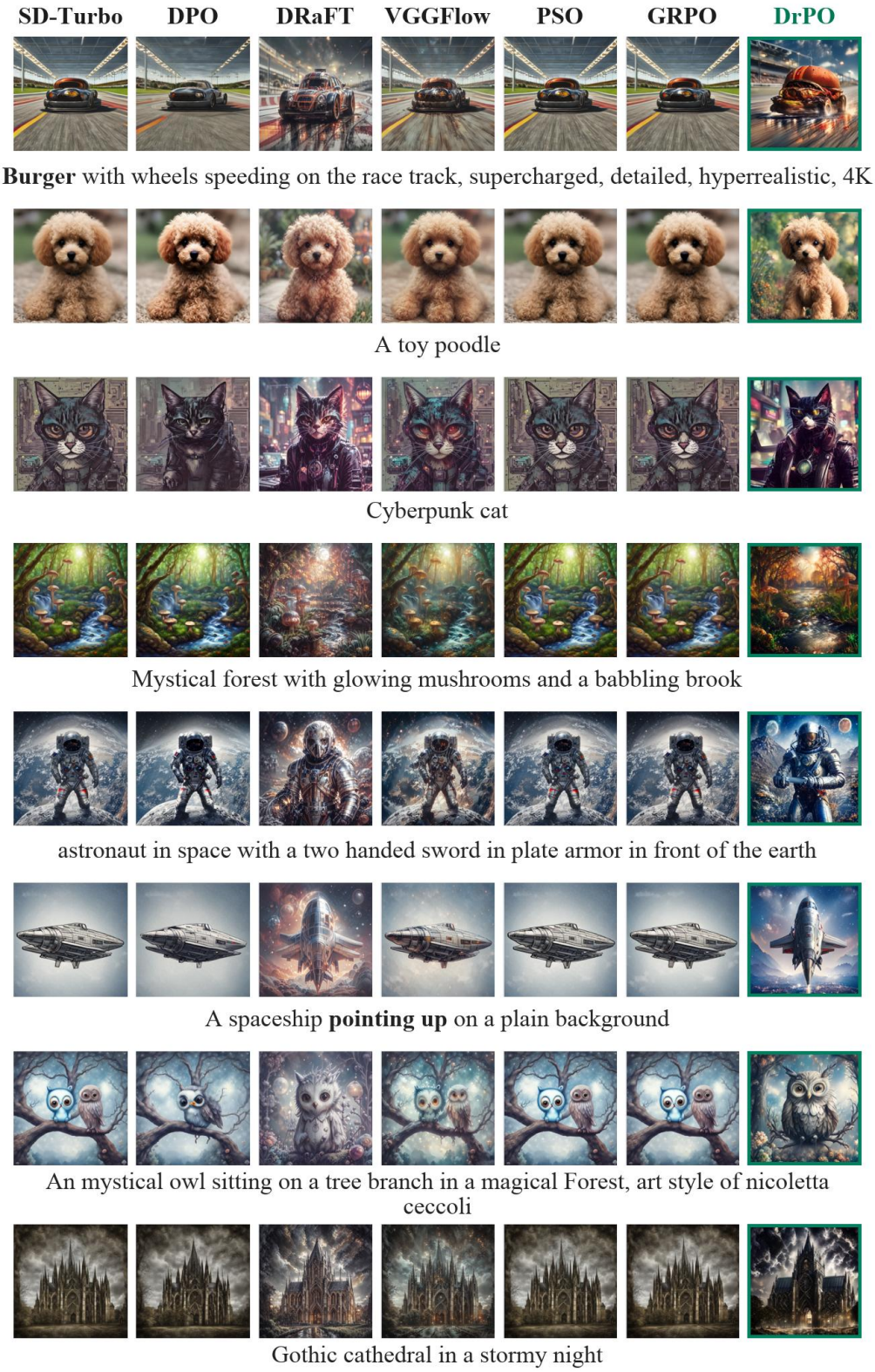


Figure A4: Full qualitative comparison corresponding to Figure 7. The grid uses a larger selected prompt set with matched seed 42.

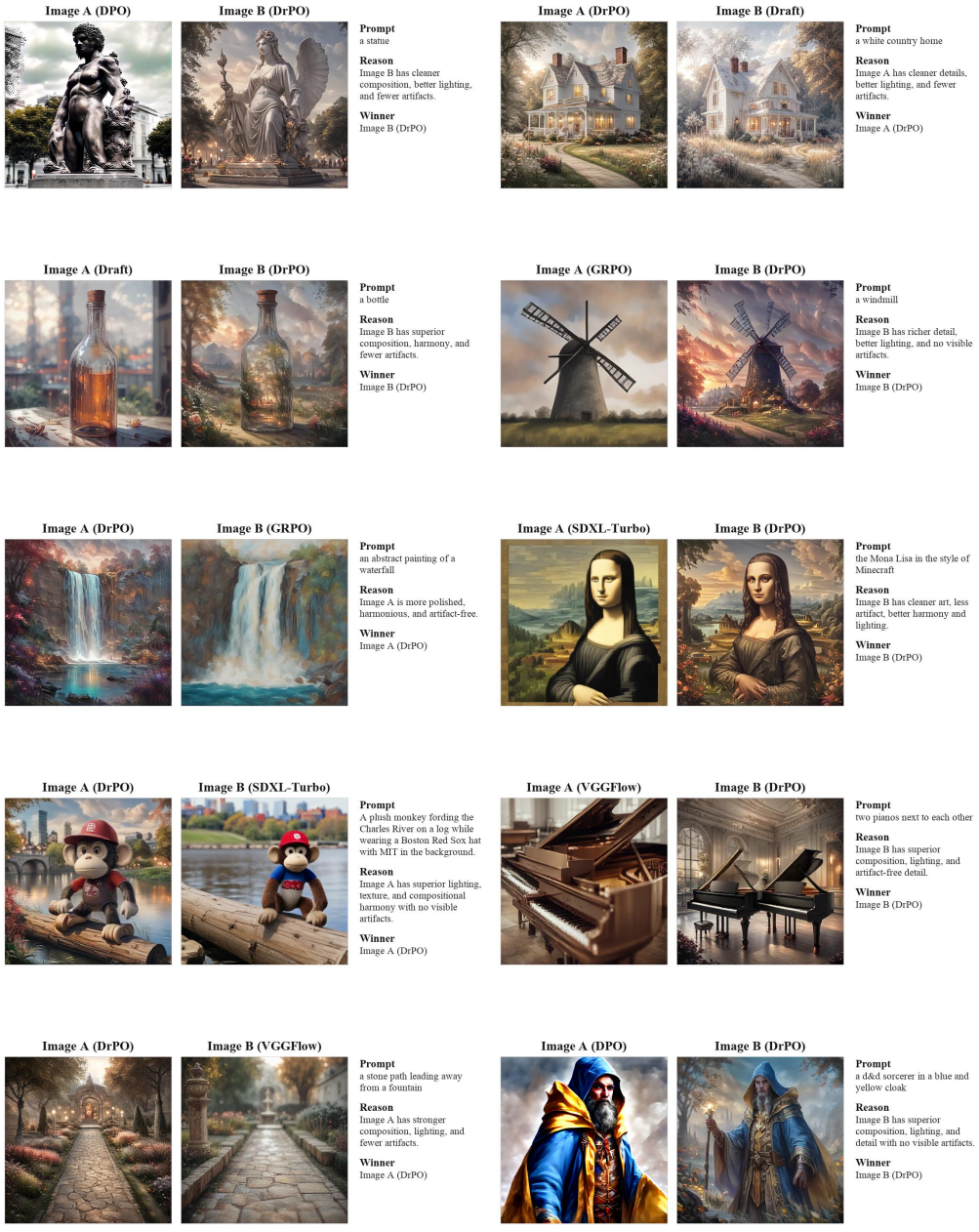


Figure A5: Pairwise examples from the Qwen3-VL presentation-quality evaluation. Each row contains two matched-seed pairs.

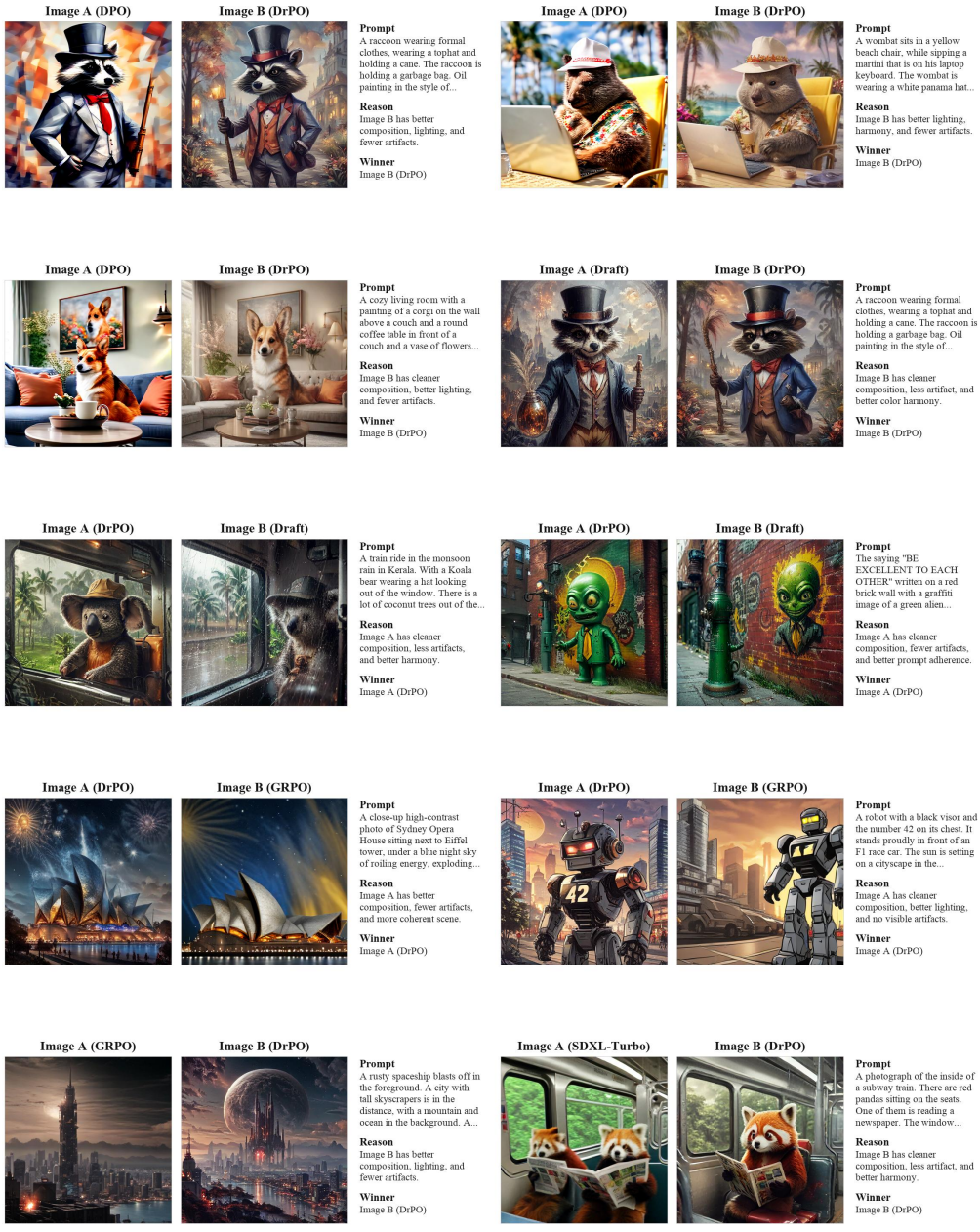


Figure A6: Additional pairwise examples from the Qwen3-VL presentation-quality evaluation.

Image A (DrPO)

Image B (SDXL-Turbo)

Prompt
A smiling sloth wearing a leather jacket, a cowboy hat, a kilt and a bowtie. The sloth is holding a quarterstaff and a big book. A shiny VW van...

Reason
Image A has better lighting, detail, and fewer artifacts.

Winner
Image A (DrPO)

Image A (SDXL-Turbo)

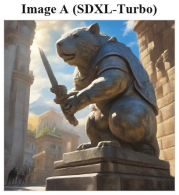


Image B (DrPO)

Prompt
A soft beam of light shines down on an armored granite wombat warrior statue holding a broad sword. The statue stands an ornate...

Reason
Image B has better lighting, detail, and harmony with prompt.

Winner
Image B (DrPO)

Image A (VGGFlow)

Image B (DrPO)

Prompt
Horses pulling a carriage on the moon's surface, with the Statue of Liberty and Great Pyramid in the background. The Planet Earth can be seen...

Reason
Image B has cleaner composition, fewer artifacts, and better harmony.

Winner
Image B (DrPO)

Image A (DrPO)

Image B (VGGFlow)

Prompt
a blue cow is standing next to a tree with red leaves and yellow fruit. the cow is standing in a field with white flowers. impressionistic...

Reason
Image A has cleaner composition, better color harmony, and fewer artifacts.

Winner
Image A (DrPO)

Image A (DrPO)

Image B (VGGFlow)

Prompt
The saying "BE EXCELLENT TO EACH OTHER" written in faded paint on the hull of an old wooden boat and reflected in...

Reason
Image A has cleaner composition, better lighting, and no visible artifacts while preserving all prompt...

Winner
Image A (DrPO)

Image A (DPO)

Image B (DrPO)

Prompt
A smooth purple octopus sitting on a rock in the middle of the sea, waves crashing, golden hour, sun reflections, high quality 3d render

Reason
Image B has cleaner composition, better lighting, and fewer artifacts.

Winner
Image B (DrPO)

Image A (DrPO)

Image B (DPO)

Prompt
A set of museum-quality emerald bracelets and beads in green in a display box at the auction, 32k, highest resolution, hyper realistic

Reason
Image A has cleaner composition, better lighting, and fewer artifacts.

Winner
Image A (DrPO)

Image A (DPO)

Image B (DrPO)

Prompt
close up of Italian pizza margherita, a glass of fresh beer, candle light, polished wood table, UHD, high quality, high detail, ultra...

Reason
Image B has cleaner composition, better lighting, and fewer artifacts.

Winner
Image B (DrPO)

Image A (Draft)

Image B (DrPO)

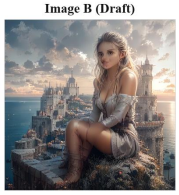
Prompt
full body, a slender anime girl with long cyan hair, powerful arcane wizard, beautiful white outfit, extremely detailed, realistic shading

Reason
Image B has cleaner composition, better lighting, and fewer artifacts.

Winner
Image B (DrPO)

Image A (DrPO)

Image B (Draft)



Prompt
A beautiful girl sitting on the top of a castle overlooking the ocean

Reason
Image A has cleaner composition, better lighting, and fewer artifacts.

Winner
Image A (DrPO)

Image A (Draft)

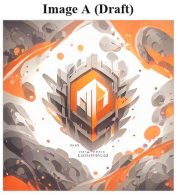
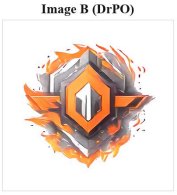


Image B (DrPO)



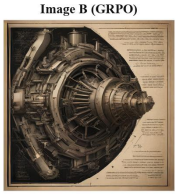
Prompt
flat logo, gaming logo, white background, vector, orange, grey,symbol, abstract, hd, 4k

Reason
Image B is preferred: Cleaner, more polished, fewer artifacts, better color harmony.

Winner
Image B (DrPO)

Image A (DrPO)

Image B (GRPO)



Prompt
a full page design of spaceship engine, black and bronze paper, intricate, highly detailed, epic, infographic, marginalia –ar 9:16...

Reason
Image A has cleaner composition, better lighting, and fewer artifacts.

Winner
Image A (DrPO)

Image A (GRPO)

Image B (DrPO)

Prompt
Creation of the world, minimalistic, abstract, mist, vector, flat, unreal engine 5, by jewel tones, scandi style, cinematic, aspect ratio 2:3, 4k

Reason
Image B has better harmony, less artifacts, and stronger cinematic quality.

Winner
Image B (DrPO)

Image A (GRPO)

Image B (DrPO)

Prompt
The Amur tiger is a rider on a Harley Davidson motorcycle. In a cool helmet, in a biker jacket. High speed. The tail flutters in the wind

Reason
Image B has superior lighting, detail, and no artifacts.

Winner
Image B (DrPO)

Image A (DrPO)

Image B (SDXL-Turbo)

Prompt
a minimalistic style D shaped logo comprises of a profile head, dominos, chess, and Inside it there is a brain shapedsocial network.

Reason
Image A is cleaner, more coherent, and better integrates all prompt elements into a unified logo design.

Winner
Image A (DrPO)

Image A (DrPO)

Image B (SDXL-Turbo)

Prompt
A set of emerald bracelets in green in a display box at the auction, uplight, very realist, very detailed, highest resolution, hyper realistic

Reason
Image A has cleaner composition, better lighting, and fewer artifacts.

Winner
Image A (DrPO)

Image A (DrPO)

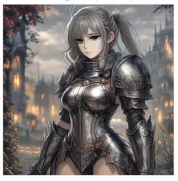


Image B (SDXL-Turbo)

Prompt
a manga beautiful girl with a medieval armor, trending on deviant art, high res, gray scales, kentaro miura style, full body.

Reason
Image A has better lighting, detail, and background harmony.

Winner
Image A (DrPO)

Image A (VGGFlow)

Image B (DrPO)

Prompt
An mystical owl sitting on a tree branch in a magical Forest, art style of nicoletta ceccoli

Reason
Image B has superior detail, harmony, and fewer artifacts.

Winner
Image B (DrPO)

Image A (DrPO)

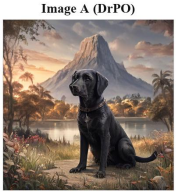
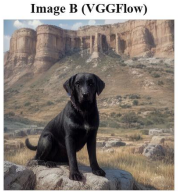


Image B (VGGFlow)



Prompt
un perro negro con el pecho y el morro blanco más pequeño que un labrador

Reason
Image A has superior composition, lighting, and harmony with no visible artifacts.

Winner
Image A (DrPO)

Image A (DrPO)

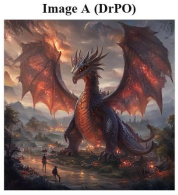


Image B (VGGFlow)

Prompt
dragon with six pairs of wings aterrorizing humans in a village near to a volcano

Reason
Image A has cleaner composition, fewer artifacts, and better harmony.

Winner
Image A (DrPO)