ALIGNING ONE-STEP GENERATIVE MODELS WITH REWARD-WEIGHTED TRANSPORT DISTILLATION

Anonymous authors
Paper under double-blind review

ABSTRACT

One-step generators enable high-quality visual generation with a single network evaluation, but their post-training is difficult: general implicit generators provide neither tractable likelihoods nor denoising trajectories, and many rewards are non-differentiable. We introduce *Reward-Weighted Transport Distillation* (RWTD), a post-training method that requires only generated samples and scalar reward evaluations. Rather than aligning solely to the conventional reward-tilted reference distribution, RWTD constructs an adaptive target that mixes separately tilted current and reference distributions. The current component incorporates improvements discovered during training, while the reference component anchors the target to the pretrained generator. RWTD realizes this target through feature-space optimal transport and fixed-point regression. Theoretical analysis shows that the fixed-point distributions of RWTD interpolate between off-policy reward tilting of the reference and on-policy tilting of the current model, providing a principled approach to balancing reward adaptation with retention of prior knowledge. Empirically, RWTD substantially improves the GenEval score of the one-step SANA Sprint 1.6B backbone from 0.73 to 0.80, while separate preference alignment experiments demonstrate strong cross-reward generalization that yields balanced improvements and preservation of compositional capabilities.

1 INTRODUCTION

Diffusion (Ho et al., 2020; Song et al., 2020; Karras et al., 2022) and flow-based models (Lipman et al., 2022; Albergo et al., 2025) achieve high-fidelity visual synthesis but typically require many sequential network evaluations. Distillation methods—including consistency models (Song et al., 2023; Luo et al., 2023), adversarial diffusion distillation (Sauer et al., 2024), and distribution matching distillation (Yin et al., 2024b;a)—compress multi-step teachers into few- or one-step generators. More recently, objectives based on drifting (Deng et al., 2026), Wasserstein gradient flows (Han et al., 2026a), mean flows (Geng et al., 2026a;b), and inductive moment matching (Zhou et al., 2025) have enabled one-step generators to be trained directly. As these models improve, developing effective post-training methods for general one-step generators becomes increasingly important.

Existing alignment methods do not yet provide a general interface. Diffusion reinforcement learning and preference optimization commonly require differentiable rewards (Clark et al., 2024), likelihood surrogates (Wallace et al., 2024; Wang et al., 2026; Xue et al., 2025), or access to multi-step denoising or flow trajectories (Black et al., 2024; Fan et al., 2023; Liu et al., 2026). Recent methods support one- and few-step generators but remain tied to diffusion dynamics (Wu et al., 2026; Luo et al., 2024), rely on high-variance reward-gradient and kernel density estimation (Lee et al., 2026), or use specialized preference-field constructions (Jiang et al., 2026). We instead seek a method that is agnostic to the generator's pre-training procedure and compatible with black-box rewards.

To this end, we introduce Reward-Weighted Transport Distillation (RWTD), whose central ingredient is an adaptive alignment target that separately tilts and mixes current and reference distributions. The current component incorporates improvements discovered during training, while the reference component anchors the target to pretrained modes. Feature-space optimal transport then determines how current samples should move toward this target, and the resulting assignments are amortized through feature regression using only scalar reward evaluations. To our knowledge, RWTD is the first one-step alignment method to explicitly target a mixture of separately normalized reward-tilted

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.

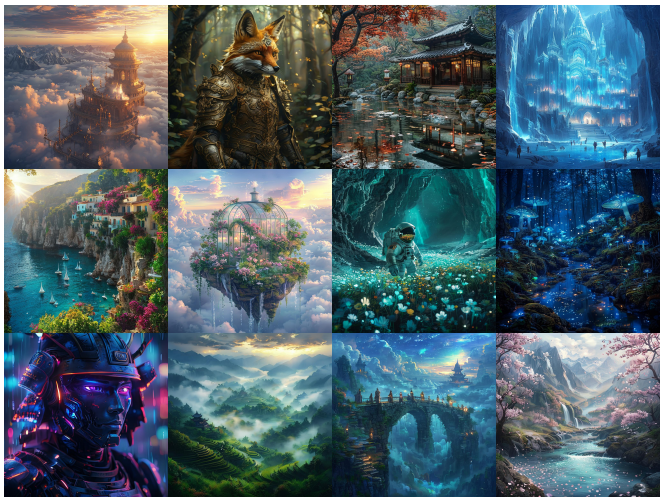


Figure 1: Images generated in one step by SANA Sprint 1.6B after RWTD post-training.

current and reference distributions. Our theoretical analysis characterizes the resulting fixed points: off-policy targeting recovers the conventional reward-tilted reference, on-policy tilting concentrates on reward-maximizing modes, and intermediate mixtures define a distinct fixed-point family with a reward-dependent rational amplification controlled by the reference mass, providing a principled trade-off between reward adaptation and reference anchoring.

Using non-differentiable compositional feedback, RWTD improves the GenEval score of one-step SANA Sprint 1.6B (Chen et al., 2025) from **0.73 to 0.80**, establishing state-of-the-art performance among SANA Sprint-based one-step generators and surpassing multi-step models of comparable scale. When trained with HPSv2 (Wu et al., 2023), RWTD also improves the optimized reward while generalizing across held-out alignment metrics. It produces more balanced gains and better preserves semantic and compositional capabilities, while gradient-based alternatives actively degrade performance on out-of-domain tasks. Detailed ablations isolate how the specific design choices of RWTD improve overall performance and data efficiency. Our main contributions are:

- **A new adaptive alignment target.** RWTD replaces reference-only tilting with a mixture of reward-tilted current and reference distributions, realized through feature-space OT using only samples and scalar (potentially black-box) rewards.
- **Interpretable theoretical analysis.** We characterize the mixed reward-tilting fixed points, revealing a reward-dependent rational amplification beyond standard reference tilting.
- **Strong compositional and preference alignment.** RWTD achieves SOTA GenEval performance at SANA Sprint 1.6B scale. Separate preference alignment experiments show strong cross-reward generalization.

2 Related Work

2.1 One-Step Generative Models

One-step generators are trained typically by distilling multi-step teachers through consistency, adversarial, or distribution-matching objectives (Song et al., 2023; Luo et al., 2023; Sauer et al., 2024; Yin et al., 2024b;a; Liu et al., 2023), or through native objectives based on drifting, Wasserstein gradient flows, mean flows, and inductive moment matching (Deng et al., 2026; Han et al., 2026a; Geng et al., 2026a;b; Zhou et al., 2025). RWTD addresses the complementary problem of aligning an existing one-step generator with a new scalar reward.

2.2 Reward Alignment of Visual Generators

Visual generators have been aligned through direct reward backpropagation (Prabhudesai et al., 2023; Clark et al., 2024), reinforcement learning (Black et al., 2024; Fan et al., 2023; Liu et al.,

2026; Zheng et al., 2026; Choi et al., 2026; Xue et al., 2025), and preference optimization (Wallace et al., 2024; Wang et al., 2026; Lu et al., 2025a;b; Han et al., 2026b). These approaches can substantially improve target rewards, but generally assume differentiable rewards, tractable likelihood surrogates, or access to multi-step denoising or flow trajectories. Complementary inference-time steering methods adapt generation through reward- or likelihood-guided sampling, search, or derivative-free posterior inference, but leave the generator unchanged and require additional sampling computation (Singhal et al., 2025; Uehara et al., 2025; Huang et al., 2026; Bansal et al., 2024; Yeh et al., 2025; Zheng et al., 2025; Wang et al., 2025). We focus on post-training alignment that amortizes reward improvement into the generator while preserving one-step inference.

Recent methods align few- and one-step generators but retain important restrictions. DI++, DI*, and DIDR (Luo, 2024; Luo et al., 2024; Wu et al., 2026) are diffusion-specific and require reward gradients, while FAV (Lee et al., 2026) requires kernel density estimation (KDE) and reward gradients. DrPO (Jiang et al., 2026) avoids gradients by constructing a rank-based dipole preference field from generated candidates. In contrast, RWTD introduces a different alignment target—an adaptive mixture of reward-tilted current and reference distributions—and realizes it through feature-space OT, without requiring reward gradients, KDE, or access to a multi-step diffusion teacher.

2.3 Optimal Transport for One-Step Generative Modeling

Optimal transport (OT) provides a geometric framework for moving probability mass between distributions, while entropic regularization enables efficient approximation through the Sinkhorn algorithm (Cuturi, 2013; Gabriel & Marco, 2019). Recently, W-Flow (Han et al., 2026a) trains a native one-step generator by defining a Wasserstein gradient flow from a reference distribution toward the data distribution and amortizing its evolution into a neural network. RWTD instead addresses post-training, using transport to realize reward-driven distributional updates for an existing generator.

3 Background and Problem Setting

3.1 Black-Box Alignment of One-Step Generators

Let G_θ be a conditional one-step generator that maps a latent variable $\boldsymbol{z} \sim p_z$ and condition $c \sim p_{\text{prompt}}$ directly to a sample $\boldsymbol{x} = G_\theta(\boldsymbol{z}, c)$, inducing the implicit conditional distribution $p_\theta(\boldsymbol{x} \mid c)$. We also assume access to a frozen reference generator G_{ref}, with distribution $p_{\text{ref}}(\boldsymbol{x} \mid c)$, and a scalar reward function $r(\boldsymbol{x}, c)$. Our objective is to optimize the reward scores of our model's samples while preserving the quality, diversity, and conditional fidelity of the reference generator.

We consider a black-box alignment setting in which p_θ and p_{ref} need not have tractable likelihoods, and r need not be differentiable. In particular, we do not assume access to a diffusion trajectory, score function, likelihood ratio, or reward gradient.

A standard operation for converting rewards into a target distribution is exponential reward tilting (Wallace et al., 2024; Uehara et al., 2024; Han et al., 2026b; Fan et al., 2023). For any conditional base distribution $p(\boldsymbol{x} \mid c)$ and inverse temperature $\beta > 0$, define

$$\begin{aligned} Z_p(c) &= \mathbb{E}_{\boldsymbol{x} \sim p(\cdot|c)} \left[\exp(\beta r(\boldsymbol{x}, c)) \right], \\ \mathcal{T}_{\beta,r}[p](\boldsymbol{x} \mid c) &= \frac{p(\boldsymbol{x} \mid c) \exp(\beta r(\boldsymbol{x}, c))}{Z_p(c)}. \end{aligned} \tag{1}$$

The inverse temperature β controls alignment strength: as $\beta \to 0$, the tilted distribution approaches p, while larger values concentrate mass on high-reward regions. Reward tilting has the variational characterization (Donsker & Varadhan, 1975)

$$\mathcal{T}_{\beta,r}[p](\cdot \mid c) = \arg\max_q \left\{ \mathbb{E}_q[r(\boldsymbol{x}, c)] - \frac{1}{\beta} \mathbb{D}_{\text{KL}}(q(\cdot \mid c) \,\|\, p(\cdot \mid c)) \right\}. \tag{2}$$

Thus, reward tilting increases expected reward while penalizing deviation from the base distribution. RWTD builds on this canonical alignment primitive by mixing tilted current and reference distributions.

3.2 Entropic Optimal Transport

Consider empirical measures $\mu = \sum_{i=1}^{N} a_i \delta_{\boldsymbol{u}_i}$ and $\nu = \sum_{j=1}^{M} b_j \delta_{\boldsymbol{v}_j}$, where $\boldsymbol{a}$ and $\boldsymbol{b}$ are probability vectors. A transport coupling between μ and ν is a nonnegative matrix whose row and column sums match these prescribed weights. The set of admissible couplings is $\Pi(\boldsymbol{a}, \boldsymbol{b}) = \{\boldsymbol{P} \in \mathbb{R}_{+}^{N \times M} : \boldsymbol{P}\mathbf{1}_M = \boldsymbol{a},\ \boldsymbol{P}^\top \mathbf{1}_N = \boldsymbol{b}\}$. Given pairwise costs C_{ij} between $\boldsymbol{u}_i$ and $\boldsymbol{v}_j$, entropic OT computes

$$\boldsymbol{P}_\epsilon = \underset{\boldsymbol{P} \in \Pi(\boldsymbol{a}, \boldsymbol{b})}{\arg\min} \left\{ \langle \boldsymbol{P}, \boldsymbol{C} \rangle + \epsilon \sum_{i,j} P_{ij} \big(\log P_{ij} - 1 \big) \right\}. \tag{3}$$

The regularization $\epsilon > 0$ smooths the coupling, while $\epsilon \to 0$ recovers unregularized OT under standard conditions. Each row of the coupling represents how much mass is assigned from a source particle to each target particle. The Sinkhorn algorithm efficiently computes this coupling by iteratively rescaling the kernel $\exp(-\boldsymbol{C}/\epsilon)$ to match the prescribed marginals (Cuturi, 2013).

4 Reward-Weighted Transport Distillation

4.1 Mixed Reward-Tilted Particle Target

The central design choice in RWTD is the target distribution. Unlike canonical alignment to a fixed reward-tilted reference, RWTD constructs a target that evolves with the current generator while retaining an explicit reference anchor. Given the current generator distribution p_θ and reference distribution p_{ref}, we define the mixed reward-tilted target as

$$Q_{\beta,\rho}[p_\theta] = (1 - \rho)\mathcal{T}_{\beta,r}[p_\theta] + \rho \mathcal{T}_{\beta,r}[p_{\text{ref}}], \tag{4}$$

where $0 \le \rho \le 1$ is a scalar hyperparameter controlling the mass allocated to the reward-tilted reference distribution. Intuitively, ρ trades off on-policy and off-policy tilting behavior. The on-policy component $\mathcal{T}_{\beta,r}[p_\theta]$ exploits high-reward regions discovered by the current generator, but evolves as the generator is updated. The off-policy component $\mathcal{T}_{\beta,r}[p_{\text{ref}}]$ instead provides a stationary source of high-reward samples from the pretrained model, continually reintroducing modes that may have been lost by the current generator. The endpoints $\rho = 0$ and $\rho = 1$ correspond to purely on-policy and purely off-policy targets, respectively.

The adaptive target in Equation 4 can be approximated with particles. For a conditioning input c, we draw independent latent variables and generate current and reference particles as

$$\begin{aligned} \boldsymbol{z}_i &\sim p_z, \quad \boldsymbol{x}_i = G_\theta(\boldsymbol{z}_i, c), \qquad i = 1, \dots, N, \\ \boldsymbol{z}'_j &\sim p_z, \quad \boldsymbol{y}_j = G_{\text{ref}}(\boldsymbol{z}'_j, c), \quad j = 1, \dots, M. \end{aligned} \tag{5}$$

We then form separately normalized reward weights for the current and reference particles:

$$\begin{aligned} w_i^\theta &= \frac{\exp(\beta r(\boldsymbol{x}_i, c))}{\sum_{\ell=1}^{N} \exp(\beta r(\boldsymbol{x}_\ell, c))}, \quad i = 1, \dots, N, \\ w_j^{\text{ref}} &= \frac{\exp(\beta r(\boldsymbol{y}_j, c))}{\sum_{\ell=1}^{M} \exp(\beta r(\boldsymbol{y}_\ell, c))}, \quad j = 1, \dots, M. \end{aligned} \tag{6}$$

The resulting empirical target measure and uniformly weighted source measure are

$$\begin{aligned} \widehat{Q}_{\beta,\rho}[p_\theta] &= (1 - \rho) \sum_{i=1}^{N} w_i^\theta \delta_{\boldsymbol{x}_i} + \rho \sum_{j=1}^{M} w_j^{\text{ref}} \delta_{\boldsymbol{y}_j}, \\ \widehat{p}_\theta &= \frac{1}{N} \sum_{i=1}^{N} \delta_{\boldsymbol{x}_i}. \end{aligned} \tag{7}$$

Separate normalization ensures that the current and reference particles receive total masses $1-\rho$ and ρ, respectively. This construction requires only generated samples and scalar reward evaluations; neither generator likelihoods nor reward gradients are needed. We treat $\widehat{p}_\theta$ as the source measure and $\widehat{Q}_{\beta,\rho}[p_\theta]$ as the target measure, and construct a feature-space optimal-transport coupling between them in the next section.

4.2 Feature-Space Optimal Transport

We construct a transport coupling from the uniformly weighted current distribution to the mixed reward-weighted target in Equation 7. Because Euclidean distance in pixel space poorly reflects perceptual and semantic similarity, we instead perform transport in the representation space of a frozen visual encoder ϕ, following standard practice (Deng et al., 2026; Han et al., 2026a).

For each current particle, let $\boldsymbol{u}_i = \phi(\boldsymbol{x}_i)$ denote its source feature. We concatenate the encoded current and reference particles into the target features by setting $\boldsymbol{v}_i = \phi(\boldsymbol{x}_i)$ for $i = 1, \dots, N$ and $\boldsymbol{v}_{N+j} = \phi(\boldsymbol{y}_j)$ for $j = 1, \dots, M$. Collecting the reward weights into vectors $\boldsymbol{w}^\theta = (w_1^\theta, \dots, w_N^\theta)^\top$ and $\boldsymbol{w}^{\mathrm{ref}} = (w_1^{\mathrm{ref}}, \dots, w_M^{\mathrm{ref}})^\top$, the source and target marginals are $\boldsymbol{a} = \mathbf{1}_N/N$ and $\boldsymbol{b} = [(1-\rho)\boldsymbol{w}^\theta; \rho \boldsymbol{w}^{\mathrm{ref}}]$, respectively. We define the feature-space cost by $C_{ik} = \|\boldsymbol{u}_i - \boldsymbol{v}_k\|_2^2$, yielding $\boldsymbol{C} \in \mathbb{R}^{N \times (N+M)}$. Using the Sinkhorn algorithm, we solve the entropic OT problem described in Section 3.2:

$$\boldsymbol{P}_\epsilon = \mathrm{Sinkhorn}_\epsilon\left(\boldsymbol{C}; \boldsymbol{a}, \boldsymbol{b}\right) \in \Pi(\boldsymbol{a}, \boldsymbol{b}). \tag{8}$$

The resulting plan satisfies $\boldsymbol{P}_\epsilon \mathbf{1}_{N+M} = \boldsymbol{a}$ and $\boldsymbol{P}_\epsilon^\top \mathbf{1}_N = \boldsymbol{b}$. Each row specifies how the mass of one current particle is distributed across the target particles. The coupling geometrically realizes the mixed target without reward gradients or diffusion-specific quantities. In the next subsection, we convert the conditional assignments into sample-specific regression targets.

4.3 Partial Transport Distillation

Each row of $\boldsymbol{P}_\epsilon$ defines a conditional distribution over mixed target particles after normalization by its mass $a_i = 1/N$. RWTD partially moves each source feature toward the conditional mean:

$$\widetilde{\boldsymbol{u}}_i = (1-\eta)\boldsymbol{u}_i + \eta \underbrace{\frac{1}{a_i} \sum_{k=1}^{N+M} P_{\epsilon,ik} \boldsymbol{v}_k}_{\bar{\boldsymbol{u}}_i}, \qquad 0 < \eta \leq 1, \tag{9}$$

where $\bar{\boldsymbol{u}}_i$ is the barycentric destination and η is the transport step size. Setting $\eta = 1$ uses the full destination, while smaller values produce more conservative updates. Since practical couplings are estimated from finite minibatches and amortized by regression, partial steps are used to limit sensitivity to coupling noise and abrupt distribution shifts.

RWTD amortizes these particle-level transport targets into the generator by regressing the features of samples generated from the original latents toward their detached targets:

$$\mathcal{L}_{\mathrm{RWTD}}(\theta) = \frac{1}{N} \sum_{i=1}^{N} \left\| \phi(G_\theta(\boldsymbol{z}_i, c)) - \mathrm{sg}(\widetilde{\boldsymbol{u}}_i) \right\|_2^2, \tag{10}$$

where $\mathrm{sg}(\cdot)$ denotes the stop-gradient operator. The encoder ϕ remains frozen, but gradients propagate through $\phi(G_\theta(\boldsymbol{z}_i, c))$ to update the generator. No gradients are taken through the reward weights, transport coupling, or target construction. Detailed algorithm pseudocode is provided in Algorithms 1 and 2.

Because the targets are recomputed from the current generator after each update, this objective constitutes a form of fixed-point regression: RWTD alternates between constructing transport targets from the current distribution and distilling those targets back into the generator. This motivates a fixed-point interpretation, made precise in the next section.

5 THEORETICAL ANALYSIS

We analyze an idealized population-level version of RWTD that isolates the distributional dynamics induced by reward tilting and mixing. We fix a condition c and suppress it from the notation.

5.1 FIXED POINTS OF MIXED REWARD TILTING

Consider the idealized full-update recursion

$$p_{t+1} = Q_{\beta,\rho}[p_t] = (1-\rho)\mathcal{T}_{\beta,r}[p_t] + \rho\mathcal{T}_{\beta,r}[p_{\mathrm{ref}}]. \tag{11}$$

We characterize fixed points of this operator and contrast it with the canonical reward-tilted reference obtained at $\rho = 1$.

Proposition 5.1 (Characterization of mixed reward-tilting fixed points)**.** *Let $0 < \rho \leq 1$ and consider fixed points satisfying $p_\infty \ll p_{\mathrm{ref}}$. Define*

$$H(\lambda) = \mathbb{E}_{\boldsymbol{x}\sim\mathcal{T}_{\beta,r}[p_{\mathrm{ref}}]}\left[\frac{1}{1-\lambda\exp(\beta r(\boldsymbol{x}))}\right]$$

on the set of nonnegative λ for which the denominator is positive $\mathcal{T}_{\beta,r}[p_{\mathrm{ref}}]$-almost everywhere and the expectation is finite. Fixed points are in one-to-one correspondence with solutions of

$$H(\lambda) = \frac{1}{\rho}. \tag{12}$$

Any such solution is unique and the corresponding fixed point is

$$p_\infty(\boldsymbol{x}) = \frac{\rho}{Z_{\mathrm{ref}}}\frac{p_{\mathrm{ref}}(\boldsymbol{x})\exp(\beta r(\boldsymbol{x}))}{1-\lambda\exp(\beta r(\boldsymbol{x}))}. \tag{13}$$

In particular, when $\rho = 1$, the unique solution is $\lambda = 0$ and $p_\infty = \mathcal{T}_{\beta,r}[p_{\mathrm{ref}}]$, recovering the standard reward-tilted reference.

The proof and further analysis are deferred to Sections A.1 and A.2. Mixed targeting is not equivalent to interpolating model parameters or merely changing the reward temperature. At $\rho = 1$, the operator recovers the conventional reward-tilted reference distribution, whereas for $0 < \rho < 1$, any fixed point contains the additional rational factor $(1-\lambda\exp(\beta r(\boldsymbol{x})))^{-1}$, defining a distinct family of distributions that preferentially amplifies high-reward regions. Decreasing ρ increases λ and sharpens this amplification.

The purely on-policy endpoint $\rho = 0$ must be treated separately because it has no unique fixed point. In this case,

$$p_t(\boldsymbol{x}) = \frac{p_0(\boldsymbol{x})\exp(t\beta r(\boldsymbol{x}))}{\mathbb{E}_{\boldsymbol{x}'\sim p_0}\left[\exp(t\beta r(\boldsymbol{x}'))\right]}. \tag{14}$$

which concentrates on the reward-maximizing regions in the support of p_0 as t increases. Thus, we view ρ as controlling the trade-off between reference anchoring and recursive reward sharpening: larger values keep the fixed point closer to the reward-tilted reference distribution, whereas smaller values give greater influence to the on-policy component and increasingly concentrate mass on high-reward modes.

5.2 FIXED-POINT INVARIANCE UNDER PARTIAL TRANSPORT

We next study whether the transport step size η changes the fixed points of the idealized population dynamics. Assume that p and $Q_{\beta,\rho}[p]$ are probability measures on $\mathbb{R}^d$ with finite second moments, with p absolutely continuous, and let T_p be the exact quadratic-cost optimal transport map from p to $Q_{\beta,\rho}[p]$, such that $(T_p)_{\#}p = Q_{\beta,\rho}[p]$. The corresponding RWTD partial transport update is

$$\mathcal{F}_\eta[p] = \left((1-\eta)\,\mathrm{Id} + \eta T_p\right)_{\#} p, \qquad 0 < \eta \leq 1, \tag{15}$$



Figure 2: Qualitative comparison of models post-trained with HPSv2. Gradient-based methods FAV and DRaFT exhibit pronounced stylization consistent with reward overoptimization, while RWTD largely preserves realism. For the second prompt, RWTD produces a geometrically coherent lake reflection, while baselines introduce spurious trees and other inconsistencies in the reflected scene.

where $\#$ denotes the pushforward of a distribution. This is the population analogue of the partial targets in Equation 9.

Proposition 5.2 (Fixed-point invariance)**.** *Under the assumptions above, for every* $0 < \eta \leq 1$*,*

$$\mathcal{F}_\eta[p] = p \quad \Longleftrightarrow \quad Q_{\beta,\rho}[p] = p. \tag{16}$$

Consequently, the set of fixed points is independent of the transport step size η*.*

The proof is deferred to Section A.3. Thus, if any nonzero partial step leaves p unchanged, then the full target must already equal p. Conversely, if $p = Q_{\beta,\rho}[p]$, the optimal transport map is the identity almost everywhere, so every partial step also leaves p unchanged. The invariance of the idealized fixed point to the transport step size motivates partial transport as a mechanism for stabilizing training through conservative updates while preserving the fixed-point set.

We emphasize that the theoretical results in this section operate under idealized conditions and therefore provide intuition about the behavior of our method, rather than strict guarantees. Practical RWTD instead uses finite minibatches, entropic OT, and barycentric projections to approximate the idealized dynamics.

6 Experiments

6.1 Experimental Setup

Dataset and Models Our experiments used two one-step generation backbones: SANA Sprint 1.6B (Chen et al., 2025) and SDXL Turbo (Sauer et al., 2024; Podell et al., 2024). For GenEval, we follow the standard post-training protocol used in prior work (Liu et al., 2026; Zheng et al., 2026): training prompts are generated from the official GenEval codebase and filtered to be disjoint from the evaluation set. For preference alignment experiments, we trained with the HPSv2 (Wu et al., 2023) reward model and prompts from the Pick-a-Pic v2 (Kirstain et al., 2023) dataset. We use the DINOv2 (Oquab et al., 2023) model as the frozen feature encoder during SANA Sprint 1.6B training, and MAE (He et al., 2022) features during SDXL Turbo training.

Baselines We compare against recent one-step alignment methods that operate directly on the one-step generator without requiring access to a multi-step diffusion teacher or denoising trajectory. These include DrPO (Jiang et al., 2026), which constructs a dipole field toward preferred feature targets, and FAV (Lee et al., 2026), which uses Stein variational gradient descent and kernel density estimation to move samples toward the reward-tilted reference. For preference alignment, we additionally include DRaFT (Clark et al., 2024), a direct reward-backpropagation baseline. All baselines are trained with at least the computational budget of RWTD to ensure fair comparison.

(a) SDXL Turbo Comparisons

Method	NFEs	Overall	Position	Counting	Color attr.	Single obj.	Colors	Two obj.
SDXL	50	0.55	0.11	0.43	0.21	0.98	0.88	0.71
LAIR-SDXL	50	0.59	**0.14**	0.40	**0.28**	**1.00**	**0.91**	**0.83**
SDXL-Turbo	1	0.55	0.09	0.48	0.20	0.99	0.87	0.68
DrPO	1	0.58	0.09	0.58	0.26	**1.00**	0.87	0.67
RWTD	1	**0.61**	**0.14**	**0.63**	0.25	**1.00**	0.89	0.75

(b) SANA Sprint 1.6B Comparisons

Method	NFEs	Overall	Position	Counting	Color attr.	Single obj.	Colors	Two obj.
FLUX.1 Dev	50	0.66	0.22	**0.74**	0.45	0.98	0.79	0.81
Playground v3	–	0.76	0.50	0.72	0.54	0.99	0.82	**0.95**
SD3.5-L	50	0.71	0.47	0.73	0.34	0.98	0.83	0.89
SANA Sprint 4-step	4	0.75	0.57	0.59	0.54	**1.00**	**0.91**	0.90
SANA Sprint 1.6B	1	0.73	0.54	0.59	0.51	0.99	0.89	0.88
DrPO	1	0.75	0.59	0.59	0.54	0.99	0.90	0.91
FAV†	1	0.73	0.53	0.60	0.50	0.99	0.89	0.89
RWTD	1	**0.80**	**0.73**	0.61	**0.60**	**1.00**	0.90	**0.95**

Table 1: GenEval results for SDXL Turbo comparisons (left) and SANA Sprint comparisons (right). †: FAV uses zeroth order gradient estimation.

Evaluation We follow the standard GenEval evaluation protocol, using the official codebase, four images per prompt, and fixed seeds and sampling settings across methods. For preference alignment evaluation, we use the PartiPrompts (Yu et al., 2022) dataset, generating five images per prompt and reporting average PickScore Kirstain et al. (2023), HPSv2 (Wu et al., 2023), Aesthetics (Schuhmann, 2022), CLIP (Radford et al., 2021), and ImageReward (Xu et al., 2023). We also report overall GenEval scores on models post-trained with HPSv2 to measure out-of-domain generalization. In general, random seeds are standardized to ensure reproducibility and fair comparison, and all evaluation protocols follow established setups (Ghosh et al., 2023; Liu et al., 2026; Wang et al., 2026).

Implementation Details We finetune LoRA adapters (Hu et al., 2021) for both generator backbones. SANA Sprint 1.6B training is run on 8 GPUs while SDXL Turbo training uses 4 GPUs. Full hyperparameter configurations, discussion of computational cost, and extended experimental details are deferred to Sections B.1 B.2, and B.3, respectively. Additional hyperparameter ablation and diversity experiments are provided in Section C.1.

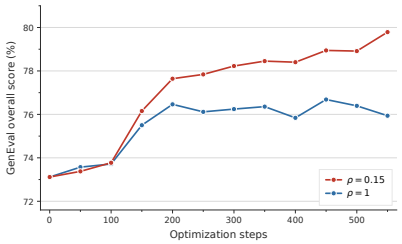
6.2 Main Experimental Results

Black-box GenEval Optimization Table 1 presents our main GenEval results. RWTD improves SDXL Turbo from 0.55 to 0.61, outperforming DrPO (0.58) and even the preference-tuned SDXL variant LAIR-SDXL (Wang et al., 2026) (50 NFEs). On SANA Sprint 1.6B, RWTD achieves 0.80, substantially improving the 0.73 base model and surpassing powerful multi-step models such as Playground v3 (Liu et al., 2024) and FLUX.1 Dev (Black Forest Labs, 2024). The largest gains occur on challenging categories such as Position and Color Attribution, demonstrating effective optimization of difficult tasks from non-differentiable scalar feedback. In contrast, DrPO and FAV provide only marginal or no improvement on this backbone, reaching 0.75 and 0.73, respectively.

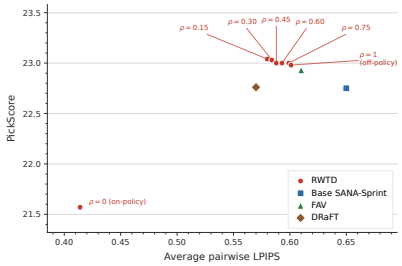
Preference Alignment and Cross-Reward Generalization We additionally present preference-alignment experiments to measure cross-reward generalization. We train on prompts from Pick-a-Pic using HPSv2 as the reward and evaluate preference metrics on PartiPrompts. RWTD shows consistently positive changes across all five reported preference metrics, whereas the competing methods trade larger gains on the optimized HPSv2 reward for decreases on some held-out metrics. While FAV and DRaFT utilize reward gradients to achieve stronger HPSv2 optimization, they produce less balanced changes across held-out metrics, both decreasing CLIP relative to the base model. The out-of-domain GenEval result is particularly informative because none of these models receive direct compositional supervision during post-training: RWTD improves overall GenEval score to 0.75, while FAV, DrPO, and DRaFT decrease it to 0.72, 0.72, and 0.62, respectively. These results suggest that RWTD is less prone to reward-specific overfitting and better preserves the generator's semantic and compositional capabilities.

Method	PS ↑	HPSv2 ↑ (in-domain)	CLIP ↑	Aes. ↑	IR ↑	GenEval ↑ (OOD)
Base model						
SANA Sprint 1.6B	22.75	30.31	0.2753	6.565	1.148	0.73
Gradient-free						
RWTD (ours)	**23.04**	32.45	**0.2761**	6.800	**1.358**	**0.75**
DrPO	22.90	32.04	0.2706	6.787	1.291	0.72
Gradient-based						
DRaFT	22.76	**37.02**	0.2573	**7.252**	1.312	0.62
FAV	22.93	35.74	0.2718	6.949	1.322	0.72

Table 2: PartiPrompt evaluation after HPSv2 post-training.



(a) Head-to-head GenEval training comparison.



(b) Held-out PickScore and diversity evaluation.

Figure 3: **Effect of mixed reward tilting. Left:** On GenEval, mixed RWTD ($\rho = 0.15$) optimizes faster and reaches a substantially higher score than the canonical off-policy target ($\rho = 1$). **Right:** Moderate reference mixing achieves the best held-out PickScore while retaining comparable perceptual diversity, whereas pure on-policy training ($\rho = 0$) substantially reduces diversity.

Mixed Targets Outperform Canonical Reference Tilting We examine whether incorporating current-model particles into the conventional reward-tilted reference target improves optimization on difficult prompts. Figure 3a compares the conventional off-policy target ($\rho = 1$) with mixed RWTD ($\rho = 0.15$). Mixing accelerates optimization and reaches 0.80 on GenEval, while off-policy RWTD plateaus near 0.77, suggesting that current-model particles are beneficial in supplying informative high-reward targets when reference samples are weak. We thus view on-policy mixing as a way to improve reward optimization and data efficiency during training.

Figure 3b evaluates the reward–diversity trade-off using average pairwise LPIPS (Zhang et al., 2018) over twenty samples for 200 PartiPrompts. Moderate mixing achieves the highest held-out PickScore with comparable diversity to competing methods, whereas pure on-policy training degrades both, reflecting overly sharp updates. Thus, on-policy particles accelerate adaptation, while reference particles prevent premature concentration. Full results appear in Section C.

Why is Optimal Transport Necessary? We isolate the role of OT by varying how per-sample targets are assigned, fixing all other hyperparameters. *RWTD-Barycentric* is our default method and uses the conditional mean in Equation 9. *RWTD-Sampled* samples a target particle from each source particle's conditional coupling row. Finally, *reward-weighted regression* (RWR) independently samples reward-weighted target particles, removing transport geometry while preserving the same target marginal. Table 3 contains full numerical results on PartiPrompts for all three variants.

Metric	Bary.	RWR	Sampled
PS	**23.02**	22.70	22.98
HPSv2	31.92	31.39	**31.96**
CLIP	0.2768	0.2733	**0.2769**
Aes.	6.751	**6.878**	6.724
IR	**1.322**	1.297	1.316

Table 3: Transport-target ablation.

The barycentric and sampled variants perform similarly across all metrics, consistent with the barycentric target being the conditional expectation of the sampled target. In contrast, independent assignment RWR better improves Aesthetics but degrades metrics like CLIP and PickScore relative to the base model. This suggests that the primary benefit comes from the sample-specific assignments induced by feature-space OT, rather than from barycentric averaging itself, and that transport geometry promotes more balanced alignment and semantic preservation.

7 Conclusion

We introduced Reward-Weighted Transport Distillation, a general post-training method for one-step generators that requires only generated samples and scalar reward evaluations. Unlike canonical reference-only tilting, RWTD mixes tilted current and reference distributions, using current particles to accelerate adaptation while reference particles help preserve diversity. RWTD thus introduces a different view of one-step generator alignment: the alignment target need not remain fixed to a reward-tilted reference distribution, but can evolve with the improving generator while retaining an explicit reference anchor. This adaptive target yields distinct fixed-point behavior and empirically improves both reward optimization and semantic preservation relative to standard reference tilting.

8 AI USE STATEMENT

In this work, we used an AI coding agent to assist with implementing the proposed methods and creating figures. All AI-assisted code was reviewed, modified as necessary, and tested by the authors. We used generative AI tools to help with grammar and paper proofreading. We did not use generative AI tools to generate synthetic datasets; develop theoretical models or conceptual frameworks; formulate mathematical claims; provide critical ingredients for or write mathematical proofs; propose or refine hypotheses; design or provide feedback on research methodology or experiments; perform translation; clean or reformat datasets; support qualitative or thematic data analysis; or interpret results. We take responsibility for the final content of this work, including all text, claims, code, and other artifacts produced with the aid of generative AI.

9 REPRODUCIBILITY STATEMENT

To support reproducibility, we provide detailed descriptions of the experimental setup and hyperparameter configurations in Appendix B. Upon publication, we will release the source code and model checkpoints needed to reproduce the reported results.

REFERENCES

Michael Albergo, Nicholas M Boffi, and Eric Vanden-Eijnden. Stochastic interpolants: A unifying framework for flows and diffusions. *Journal of Machine Learning Research*, 26(209):1–80, 2025.

Arpit Bansal, Hong-Min Chu, Avi Schwarzschild, Roni Sengupta, Micah Goldblum, Jonas Geiping, and Tom Goldstein. Universal guidance for diffusion models. In *International Conference on Learning Representations*, volume 2024, pp. 51304–51323, 2024.

Kevin Black, Michael Janner, Yilun Du, Ilya Kostrikov, and Sergey Levine. Training diffusion models with reinforcement learning. In *International Conference on Learning Representations*, volume 2024, pp. 4965–4987, 2024.

Black Forest Labs. FLUX. `https://github.com/black-forest-labs/flux`, 2024. GitHub repository.

Junsong Chen, Shuchen Xue, Yuyang Zhao, Jincheng Yu, Sayak Paul, Junyu Chen, Han Cai, Song Han, and Enze Xie. Sana-sprint: One-step diffusion with continuous-time consistency distillation. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 16185–16195. IEEE, 2025.

Jaemoo Choi, Yuchen Zhu, Wei Guo, Petr Molodyk, Bo Yuan, Jinbin Bai, Yi Xin, Molei Tao, and Yongxin Chen. Rethinking the design space of reinforcement learning for diffusion models: On the importance of likelihood estimation beyond loss design. *arXiv preprint arXiv:2602.04663*, 2026.

Kevin Clark, Paul Vicol, Kevin Swersky, and David Fleet. Directly fine-tuning diffusion models on differentiable rewards. In *International Conference on Learning Representations*, volume 2024, pp. 4793–4822, 2024.

Marco Cuturi. Sinkhorn distances: Lightspeed computation of optimal transport. *Advances in neural information processing systems*, 26, 2013.

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

Monroe D. Donsker and S. R. Srinivasa Varadhan. Asymptotic evaluation of certain markov process expectations for large time, i. *Communications on Pure and Applied Mathematics*, 28(1):1–47, 1975. doi: 10.1002/cpa.3160280102.

Ying Fan, Olivia Watkins, Yuqing Du, Hao Liu, Moonkyung Ryu, Craig Boutilier, Pieter Abbeel, Mohammad Ghavamzadeh, Kangwook Lee, and Kimin Lee. Dpok: Reinforcement learning for fine-tuning text-to-image diffusion models. *Advances in neural information processing systems*, 36:79858–79885, 2023.

Lan Feng, Wuyang Li, Eloi Zablocki, Matthieu Cord, and Alexandre Alahi. Representation distribution matching for one-step visual generation. *arXiv preprint arXiv:2607.02375*, 2026.

Peyré Gabriel and Cuturi Marco. Computational optimal transport with applications to data sciences. *Foundations and trends® in machine learning*, 11(5-6):355–607, 2019.

Zhengyang Geng, Mingyang Deng, Xingjian Bai, Zico Kolter, and Kaiming He. Mean flows for one-step generative modeling. *Advances in Neural Information Processing Systems*, 38:75460–75482, 2026a.

Zhengyang Geng, Yiyang Lu, Zongze Wu, Eli Shechtman, J Zico Kolter, and Kaiming He. Improved mean flows: On the challenges of fastforward generative models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 30467–30476, 2026b.

Dhruba Ghosh, Hannaneh Hajishirzi, and Ludwig Schmidt. Geneval: An object-focused framework for evaluating text-to-image alignment. *Advances in Neural Information Processing Systems*, 36: 52132–52152, 2023.

Jiaqi Han, Puheng Li, Qiushan Guo, Renyuan Xu, Stefano Ermon, and Emmanuel J Candès. One-step generative modeling via wasserstein gradient flows. *arXiv preprint arXiv:2605.11755*, 2026a.

Jiaqi Han, Austin Wang, Minkai Xu, Wenda Chu, Meihua Dang, Haotian Ye, Huayu Chen, Yisong Yue, and Stefano Ermon. Discrete diffusion trajectory alignment via stepwise decomposition. In *International Conference on Learning Representations*, volume 2026, pp. 5023–5047, 2026b.

Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *2022 IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, pp. 15979–15988. IEEE, 2022.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.

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. *arXiv preprint arXiv:2106.09685*, 2021.

Jerry Y Huang, Justin Lin, Sheel Shah, Kartik Nair, and Nicholas M Boffi. How to guide your flow: Few-step alignment via flow map reward guidance. *arXiv preprint arXiv:2604.27147*, 2026.

Hoseong Hwang, Woorim Han, Joungin Chun, Jinseong Park, and Jaewoong Choi. Reward-guided fine-tuning of one-step generative models via wasserstein gradient flow. *arXiv preprint arXiv:2608.29647*, 2026.

Zhou Jiang, Yandong Wen, and Zhen Liu. Drifting preference optimization for one-step generative models. *arXiv preprint arXiv:2606.02521*, 2026.

Tero Karras, Miika Aittala, Timo Aila, and Samuli Laine. Elucidating the design space of diffusion-based generative models. *Advances in neural information processing systems*, 35:26565–26577, 2022.

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. *Advances in neural information processing systems*, 36:36652–36663, 2023.

Jaewoo Lee, Hyeongyu Kang, Dohyun Kim, Kyuil Sim, Woocheol Shin, Minsu Kim, Taeyoung Yun, Jeongjae Lee, Sanghyeok Choi, Tabitha Edith Lee, et al. Aligning few-step generative models by amortizing sample-based variational inference. *arXiv preprint arXiv:2605.26552*, 2026.

Yaron Lipman, Ricky TQ Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. *arXiv preprint arXiv:2210.02747*, 2022.

Bingchen Liu, Ehsan Akhgari, Alexander Visheratin, Aleks Kamko, Linmiao Xu, Shivam Shrirao, Chase Lambert, Joao Souza, Suhail Doshi, and Daiqing Li. Playground v3: Improving text-to-image alignment with deep-fusion large language models. *arXiv preprint arXiv:2409.10695*, 2024.

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. *Advances in neural information processing systems*, 38:40783–40818, 2026.

Xingchao Liu, Xiwen Zhang, Jianzhu Ma, Jian Peng, et al. Instaflow: One step is enough for high-quality diffusion-based text-to-image generation. In *The twelfth international conference on learning representations*, 2023.

Yunhong Lu, Qichao Wang, Hengyuan Cao, Xierui Wang, Xiaoyin Xu, and Min Zhang. Inpo: Inversion preference optimization with reparametrized ddim for efficient diffusion model alignment. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 28629–28639. IEEE, 2025a.

Yunhong Lu, Qichao Wang, Hengyuan Cao, Xiaoyin Xu, and Min Zhang. Smoothed preference optimization via renoise inversion for aligning diffusion models with varied human preferences. *arXiv preprint arXiv:2506.02698*, 2025b.

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.

Weijian Luo. Diff-instruct++: Training one-step text-to-image generator model to align with human preferences. *arXiv preprint arXiv:2410.18881*, 2024.

Weijian Luo, Debing Zhang, Zhengyang Geng, et al. Diff-instruct*: Towards human-preferred one-step text-to-image generative models. 2024.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023.

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 *International Conference on Learning Representations*, volume 2024, pp. 1862–1874, 2024.

Mihir Prabhudesai, Anirudh Goyal, Deepak Pathak, and Katerina Fragkiadaki. Aligning text-to-image diffusion models with reward backpropagation. 2023.

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 *International conference on machine learning*, pp. 8748–8763. PmLR, 2021.

Axel Sauer, Dominik Lorenz, Andreas Blattmann, and Robin Rombach. Adversarial diffusion distillation. In *European Conference on Computer Vision*, pp. 87–103. Springer, 2024.

Christoph Schuhmann. Improved aesthetic predictor. `https://github.com/christophschuhmann/improved-aesthetic-predictor`, 2022. CLIP-based aesthetic score predictor.

Raghav Singhal, Zachary Horvitz, Ryan Teehan, Mengye Ren, Zhou Yu, Kathleen McKeown, and Rajesh Ranganath. A general framework for inference-time scaling and steering of diffusion models. *arXiv preprint arXiv:2501.06848*, 2025.

Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. *arXiv preprint arXiv:2011.13456*, 2020.

Yang Song, Prafulla Dhariwal, Mark Chen, and Ilya Sutskever. Consistency models. *arXiv preprint arXiv:2303.01469*, 2023.

Masatoshi Uehara, Yulai Zhao, Tommaso Biancalani, and Sergey Levine. Understanding reinforcement learning-based fine-tuning of diffusion models: A tutorial and review. *arXiv preprint arXiv:2407.13734*, 2024.

Masatoshi Uehara, Yulai Zhao, Chenyu Wang, Xiner Li, Aviv Regev, Sergey Levine, and Tommaso Biancalani. Inference-time alignment in diffusion models with reward-guided generation: Tutorial and review. *arXiv preprint arXiv:2501.09685*, 2025.

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 *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 8228–8238. IEEE, 2024.

Austin Wang, Hongkai Zheng, Zihui Wu, Ricardo Baptista, Daniel Zhengyu Huang, and Yisong Yue. Ensemble kalman sampling and diffusion prior in tandem: A split gibbs framework. In *Frontiers in Probabilistic Inference: Learning meets Sampling*, 2025.

Austin Wang, Jiaqi Han, Stefano Ermon, and Yisong Yue. Beyond pairwise preferences: Listwise reward-aware alignment for diffusion models. *arXiv preprint arXiv:2605.26491*, 2026.

Junyi Wu, Weijian Luo, Haoyang Zheng, Ruizhe Zhang, and Guang Lin. Diff-instruct with diffused reward: Towards principled one-step generator rl. *arXiv preprint arXiv:2605.24001*, 2026.

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.

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. *Advances in Neural Information Processing Systems*, 36:15903–15935, 2023.

Shuchen Xue, Chongjian Ge, Shilong Zhang, Yichen Li, and Zhi-Ming Ma. Advantage weighted matching: Aligning rl with pretraining in diffusion models. *arXiv preprint arXiv:2509.25050*, 2025.

Po-Hung Yeh, Kuang-Huei Lee, and Jun-Cheng Chen. Training-free diffusion model alignment with sampling demons. In *International Conference on Learning Representations*, volume 2025, pp. 95708–95744, 2025.

Tianwei Yin, Michaël Gharbi, Taesung Park, Richard Zhang, Eli Shechtman, Fredo Durand, and William T Freeman. Improved distribution matching distillation for fast image synthesis. *Advances in neural information processing systems*, 37:47455–47487, 2024a.

Tianwei Yin, Michaël Gharbi, Richard Zhang, Eli Shechtman, Fredo Durand, William T Freeman, and Taesung Park. One-step diffusion with distribution matching distillation. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 6613–6623. IEEE, 2024b.

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. *arXiv preprint arXiv:2206.10789*, 2(3):5, 2022.

Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *2018 IEEE/CVF conference on computer vision and pattern recognition*, pp. 586–595. IEEE, 2018.

Hongkai Zheng, Austin Wang, Zihui Wu, Zhengyu Huang, Ricardo Baptista, and Yisong Yue. Blade: A derivative-free bayesian inversion method using diffusion priors. *arXiv preprint arXiv:2510.10968*, 2025.

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*, volume 2026, pp. 134129–134150, 2026.

Linqi Zhou, Stefano Ermon, and Jiaming Song. Inductive moment matching. *arXiv preprint arXiv:2503.07565*, 2025.

A Proofs and Extended Theoretical Analysis

A.1 Derivation of the Mixed Reward-Tilting Fixed Point

We prove Proposition 5.1. Define

$$s(\boldsymbol{x}) = \exp(\beta r(\boldsymbol{x}))$$

and

$$Z_{\mathrm{ref}} = \mathbb{E}_{\boldsymbol{x}\sim p_{\mathrm{ref}}}\left[s(\boldsymbol{x})\right], \qquad q_{\mathrm{ref}}(\boldsymbol{x}) = \frac{p_{\mathrm{ref}}(\boldsymbol{x})s(\boldsymbol{x})}{Z_{\mathrm{ref}}} = \mathcal{T}_{\beta,r}[p_{\mathrm{ref}}](\boldsymbol{x}). \tag{17}$$

We first show that every reference-dominated fixed point induces a solution of Equation 12. Let $p_\infty \ll p_{\mathrm{ref}}$ be a fixed point of $Q_{\beta,\rho}$ and define

$$Z_\infty = \mathbb{E}_{\boldsymbol{x}\sim p_\infty}\left[s(\boldsymbol{x})\right], \qquad \lambda = \frac{1-\rho}{Z_\infty}. \tag{18}$$

The fixed-point condition is

$$p_\infty(\boldsymbol{x}) = (1-\rho)\frac{p_\infty(\boldsymbol{x})s(\boldsymbol{x})}{Z_\infty} + \rho q_{\mathrm{ref}}(\boldsymbol{x}). \tag{19}$$

Substituting Equation 18 and collecting the terms involving p_∞ gives

$$p_\infty(\boldsymbol{x})\left(1-\lambda s(\boldsymbol{x})\right) = \rho q_{\mathrm{ref}}(\boldsymbol{x}). \tag{20}$$

Because Equation 19 implies $p_\infty \geq \rho q_{\mathrm{ref}}$, the factor $1-\lambda s(\boldsymbol{x})$ is positive q_{ref}-almost everywhere. Therefore,

$$p_\infty(\boldsymbol{x}) = \frac{\rho q_{\mathrm{ref}}(\boldsymbol{x})}{1-\lambda s(\boldsymbol{x})} = \frac{\rho}{Z_{\mathrm{ref}}}\frac{p_{\mathrm{ref}}(\boldsymbol{x})\exp(\beta r(\boldsymbol{x}))}{1-\lambda\exp(\beta r(\boldsymbol{x}))}. \tag{21}$$

Normalizing Equation 21 yields

$$1 = \rho\mathbb{E}_{\boldsymbol{x}\sim q_{\mathrm{ref}}}\left[\frac{1}{1-\lambda s(\boldsymbol{x})}\right] = \rho H(\lambda). \tag{22}$$

Hence

$$H(\lambda) = \frac{1}{\rho},$$

so every reference-dominated fixed point induces a solution of Equation 12.

We next show uniqueness. If $0 \leq \lambda_1 < \lambda_2$ are both in the admissible domain of H, then, since $s(\boldsymbol{x}) > 0$,

$$\frac{1}{1-\lambda_1 s(\boldsymbol{x})} < \frac{1}{1-\lambda_2 s(\boldsymbol{x})}$$

q_{ref}-almost everywhere. Therefore, H is strictly increasing on its admissible domain, so $H(\lambda) = 1/\rho$ has at most one solution.

It remains to prove the converse. Suppose that an admissible $\lambda \geq 0$ satisfies

$$H(\lambda) = \frac{1}{\rho},$$

and define

$$p_\lambda(\boldsymbol{x}) = \frac{\rho q_{\mathrm{ref}}(\boldsymbol{x})}{1-\lambda s(\boldsymbol{x})}. \tag{23}$$

The defining equation for H ensures that p_λ integrates to one. Moreover,

$$\lambda\mathbb{E}_{\boldsymbol{x}\sim p_\lambda}[s(\boldsymbol{x})] = \rho\mathbb{E}_{\boldsymbol{x}\sim q_{\mathrm{ref}}}\left[\frac{\lambda s(\boldsymbol{x})}{1-\lambda s(\boldsymbol{x})}\right] \tag{24}$$

$$= \rho\left(H(\lambda)-1\right) = 1-\rho. \tag{25}$$

Thus, writing $Z_\lambda = \mathbb{E}_{\boldsymbol{x}\sim p_\lambda}[s(\boldsymbol{x})]$, we have $\lambda Z_\lambda = 1-\rho$. Combining this identity with Equation 23 gives

$$Q_{\beta,\rho}[p_\lambda](\boldsymbol{x}) = (1-\rho)\frac{p_\lambda(\boldsymbol{x})s(\boldsymbol{x})}{Z_\lambda} + \rho q_{\mathrm{ref}}(\boldsymbol{x}) \tag{26}$$

$$= \lambda s(\boldsymbol{x})p_\lambda(\boldsymbol{x}) + \rho q_{\mathrm{ref}}(\boldsymbol{x}) \tag{27}$$

$$= p_\lambda(\boldsymbol{x}). \tag{28}$$

Therefore, every admissible solution of Equation 12 defines a reference-dominated fixed point. Together with the strict monotonicity of H, this proves the claimed one-to-one correspondence and uniqueness.

Finally, when $\rho = 1$, Equation 12 becomes $H(\lambda) = 1$. Since $H(0) = 1$ and H is strictly increasing, the unique solution is $\lambda = 0$, yielding

$$p_\infty = q_{\mathrm{ref}} = \mathcal{T}_{\beta,r}[p_{\mathrm{ref}}].$$

□

A.2 Existence, Uniqueness, and Effect of the Mixing Coefficient

Proposition 5.1 shows that reference-dominated fixed points are in one-to-one correspondence with solutions of

$$H(\lambda) = \frac{1}{\rho},$$

and that such a solution is unique whenever it exists. We now give a sufficient condition ensuring existence for every $0 < \rho \leq 1$ and analyze how the resulting fixed point varies with ρ.

We continue to restrict attention to distributions absolutely continuous with respect to p_{ref}. This is a natural invariant class: if $p_0 \ll p_{\mathrm{ref}}$, then $Q_{\beta,\rho}[p_t] \ll p_{\mathrm{ref}}$ for every finite t.

Assume that $s(\boldsymbol{x}) = \exp(\beta r(\boldsymbol{x}))$ is essentially bounded under q_{ref}, and define

$$s_{\max} = \operatorname*{ess\,sup}_{\boldsymbol{x}\sim q_{\mathrm{ref}}} s(\boldsymbol{x}). \tag{29}$$

Recall that

$$H(\lambda) = \mathbb{E}_{\boldsymbol{x}\sim q_{\mathrm{ref}}}\left[\frac{1}{1-\lambda s(\boldsymbol{x})}\right], \qquad 0 \leq \lambda < s_{\max}^{-1}. \tag{30}$$

On this interval, H is continuous and strictly increasing, with

$$H'(\lambda) = \mathbb{E}_{\boldsymbol{x}\sim q_{\mathrm{ref}}}\left[\frac{s(\boldsymbol{x})}{(1-\lambda s(\boldsymbol{x}))^2}\right] > 0. \tag{31}$$

Existence. Assume additionally that

$$\lim_{\lambda\uparrow s_{\max}^{-1}} H(\lambda) = \infty. \tag{32}$$

Since $H(0) = 1$, continuity and strict monotonicity imply that H maps $[0, s_{\max}^{-1})$ bijectively onto $[1, \infty)$. Therefore, for every $0 < \rho \leq 1$, there exists a unique $\lambda_\rho \in [0, s_{\max}^{-1})$ satisfying

$$H(\lambda_\rho) = \frac{1}{\rho}. \tag{33}$$

For $0 < \rho < 1$, we have $\lambda_\rho > 0$, while for $\rho = 1$, $\lambda_\rho = 0$.

By Proposition 5.1, this solution defines the unique reference-dominated fixed point

$$p_\rho(\boldsymbol{x}) = \frac{\rho q_{\mathrm{ref}}(\boldsymbol{x})}{1-\lambda_\rho s(\boldsymbol{x})}. \tag{34}$$

The boundary condition in Equation 32 holds automatically when $\mathcal{X}$ is finite and $p_{\mathrm{ref}}(\boldsymbol{x}) > 0$ for every $\boldsymbol{x} \in \mathcal{X}$. Indeed, q_{ref} assigns positive mass to a state attaining $s_{\max}$, whose contribution to $H(\lambda)$ diverges as $\lambda \uparrow s_{\max}^{-1}$.

How does ρ sharpen the tilt? We see from Equations 30 and 31 that λ is a strictly decreasing function of ρ. Indeed, implicit differentiation yields

$$\frac{d\lambda}{d\rho} = -\frac{1}{\rho^2 H'(\lambda)} < 0. \tag{35}$$

Thus, decreasing ρ increases λ. At the off-policy endpoint $\rho = 1$, we have $\lambda = 0$ and recover ordinary reference tilting. If $H(\lambda)$ diverges as $\lambda \uparrow s_{\max}^{-1}$, then

$$\lim_{\rho \downarrow 0} \lambda(\rho) = s_{\max}^{-1}. \tag{36}$$

Finally, increasing λ sharpens the fixed point toward high-reward samples. To see this, consider two samples $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$ with $q_{\mathrm{ref}}(\boldsymbol{x}_i) > 0$ and $r(\boldsymbol{x}_1) > r(\boldsymbol{x}_2)$, and write $s_i = s(\boldsymbol{x}_i)$. Then

$$\frac{p_\rho(\boldsymbol{x}_1)/q_{\mathrm{ref}}(\boldsymbol{x}_1)}{p_\rho(\boldsymbol{x}_2)/q_{\mathrm{ref}}(\boldsymbol{x}_2)} = \frac{1-\lambda_\rho s_2}{1-\lambda_\rho s_1}. \tag{37}$$

The logarithmic derivative of this ratio is

$$\frac{d}{d\lambda} \log\left(\frac{1-\lambda s_2}{1-\lambda s_1}\right) = \frac{s_1 - s_2}{(1-\lambda s_1)(1-\lambda s_2)} > 0. \tag{38}$$

Therefore, as ρ decreases and λ increases, the fixed point assigns progressively greater relative mass to higher-reward samples. The mixing coefficient ρ thus interpolates between ordinary reference reward tilting at $\rho = 1$ and increasingly sharp on-policy reward amplification as ρ approaches zero.

A.3 Proof of Fixed-Point Invariance

We prove Proposition 5.2. For a distribution p, let

$$q_p = Q_{\beta,\rho}[p], \tag{39}$$

and let T_p be the exact quadratic-cost optimal transport map from p to q_p, so that $(T_p)_{\#}p = q_p$. Define the interpolating map

$$S_{p,\eta}(\boldsymbol{x}) = (1-\eta)\boldsymbol{x} + \eta T_p(\boldsymbol{x}), \tag{40}$$

such that the partial update is $\mathcal{F}_\eta[p] = (S_{p,\eta})_{\#}p$.

Let

$$D = W_2(p, q_p) = \left(\mathbb{E}_{\boldsymbol{x}\sim p}\left[\|T_p(\boldsymbol{x}) - \boldsymbol{x}\|_2^2\right]\right)^{1/2}. \tag{41}$$

The joint distribution of $(\boldsymbol{x}, S_{p,\eta}(\boldsymbol{x}))$ for $\boldsymbol{x} \sim p$ is a valid coupling between p and $\mathcal{F}_\eta[p]$. Therefore,

$$W_2(p, \mathcal{F}_\eta[p]) \le \left(\mathbb{E}_{\boldsymbol{x}\sim p}\left[\|S_{p,\eta}(\boldsymbol{x}) - \boldsymbol{x}\|_2^2\right]\right)^{1/2} \tag{42}$$

$$= \eta D. \tag{43}$$

Similarly, the joint distribution of $(S_{p,\eta}(\boldsymbol{x}), T_p(\boldsymbol{x}))$ is a valid coupling between $\mathcal{F}_\eta[p]$ and q_p, which gives

$$W_2(\mathcal{F}_\eta[p], q_p) \le (1-\eta)D. \tag{44}$$

Applying the triangle inequality for W_2 and combining Equations 43 and 44, we obtain

$$\begin{aligned} D &= W_2(p, q_p) && (45)\\ &\leq W_2(p, \mathcal{F}_\eta[p]) + W_2(\mathcal{F}_\eta[p], q_p) && (46)\\ &\leq \eta D + (1-\eta)D = D. && (47) \end{aligned}$$

All inequalities must therefore be equalities. In particular,

$$W_2(p, \mathcal{F}_\eta[p]) = \eta W_2(p, q_p). \tag{48}$$

We now prove the two directions of Equation 16.

First, suppose that $\mathcal{F}_\eta[p] = p$ for some $\eta > 0$. Then Equation 48 implies

$$0 = W_2(p, \mathcal{F}_\eta[p]) = \eta W_2(p, q_p). \tag{49}$$

Since $\eta > 0$, we must have $W_2(p, q_p) = 0$, and hence $p = q_p = Q_{\beta,\rho}[p]$.

Conversely, suppose that $Q_{\beta,\rho}[p] = p$. Then $q_p = p$, so the optimal quadratic transport cost is zero. Consequently, $T_p(\boldsymbol{x}) = \boldsymbol{x}$ for p-almost every $\boldsymbol{x}$, and therefore

$$S_{p,\eta}(\boldsymbol{x}) = (1-\eta)\boldsymbol{x} + \eta T_p(\boldsymbol{x}) = \boldsymbol{x} \tag{50}$$

for p-almost every $\boldsymbol{x}$. It follows that $\mathcal{F}_\eta[p] = p$ for every $0 < \eta \leq 1$.

Thus,

$$\mathcal{F}_\eta[p] = p \quad \Longleftrightarrow \quad Q_{\beta,\rho}[p] = p \tag{51}$$

for every $0 < \eta \leq 1$, proving that the fixed-point set is independent of the partial transport step size. The exclusion of $\eta = 0$ is necessary because $\mathcal{F}_0[p] = p$ for every distribution p. □

The invariance of the idealized fixed point to the partial transport step size thus motivates the usage of partial transport as a means to ensure stable training with conservative updates, without changing the target distribution.

B Extended Experimental Details

B.1 Hyperparameters

Detailed hyperparameter configurations are presented in Table 4. For each batch element, the Sinkhorn regularization is chosen adaptively as

$$\varepsilon = \max(s_{\mathrm{OT}}\ \mathrm{median}_{i,j}\, C_{ij}, \varepsilon_{\min})\,,$$

where C_{ij} is the weighted feature-space transport cost, s_{OT} is the configured regularization scale, and $\varepsilon_{\min}$ prevents numerical instability. In practice, we found that this led to stable Sinkhorn iterations, with the regularization parameter usually falling in the range $0.03 \leq \epsilon \leq 0.05$.

B.2 Computational Cost and Training Efficiency

Table 5 compares the end-to-end computational cost of post-training SANA Sprint 1.6B under identical hardware and effective particle batch sizes. On GenEval, RWTD requires 197 GPU-hours, comparable to the 194 GPU-hours required by DrPO, despite using twice as many scalar reward evaluations per step. In contrast, FAV requires 431 GPU-hours, approximately $2.2\times$ the cost of RWTD. This increase reflects the additional reward evaluations required by its zeroth-order estimator for the non-differentiable GenEval reward.

(a) SDXL-Turbo

Hyperparameter	GenEval post-training
Optimization	
LoRA rank	16
LoRA α	16
AdamW weight decay	0.0
AdamW β_1	0.9
AdamW β_2	0.999
Learning rate	5e-5
Number of GPUs	4
Gradient accumulation steps	8
Per-GPU batch size	1
Number of optimization steps	300
RWTD	
Reward temperature β	4.0
Reference mass ρ	0.15
Partial transport step η	0.20
Current particles N	24
Reference particles M	24
Feature encoder	MAE
Entropic OT scale s_{OT}	0.1
Entropic OT minimum ϵ_{min}	0.0001
Sinkhorn iterations L_{SK}	100

(b) SANA Sprint 1.6B

Hyperparameter	GenEval post-training	HPSv2 post-training
Optimization		
LoRA rank	32	32
LoRA α	32	32
AdamW weight decay	0.0	0.0
AdamW β_1	0.9	0.9
AdamW β_2	0.999	0.999
Learning rate	3e-5	1e-4
Number of GPUs	8	8
Gradient accumulation steps	4	4
Per-GPU batch size	1	1
Number of optimization steps	550	400
RWTD		
Reward temperature β	2.0	5.0
Reference mass ρ	0.15	0.15
Partial transport step η	0.20	0.20
Current particles N	24	24
Reference particles M	24	24
Feature encoder	DINOv2	DINOv2
Entropic OT scale s_{OT}	0.1	0.1
Entropic OT minimum ϵ_{min}	0.0001	0.0001
Sinkhorn iterations L_{SK}	100	100

Table 4: Optimization and RWTD hyperparameters for SDXL-Turbo and SANA Sprint 1.6B. SANA settings are reported separately for GenEval and HPSv2 post-training.

Reward	Method	GPUs	Effective batch (particles)	Reward calls per step	Reward-gradient backprop.	Steps	GPU-hours
GenEval	RWTD	8	1536	1536	No	550	197
	DrPO	8	1536	768	No	600	194
	FAV	8	1536	6144	No	550	431
HPSv2	RWTD	8	1536	1536	No	400	121
	DrPO	8	1536	768	No	400	119
	FAV	8	1536	768	Yes	400	123
	DRaFT	8	1536	1536	Yes	400	315

Table 5: Computational cost of post-training SANA Sprint 1.6B. All methods use identical hardware and effective batch sizes. Reward calls count scalar reward evaluations, while reward-gradient backpropagation indicates whether gradients are propagated through the reward model. GPU-hours measure total wall clock runtime multiplied by the number of GPUs.

The costs are generally close under HPSv2 post-training: RWTD requires 121 GPU-hours, compared with 119 for DrPO, 123 for FAV, and 315 for DRaFT. Here, FAV and DRaFT can differentiate through the HPSv2 reward, therefore avoiding the multiple evaluations required for zeroth-order estimation, although incurring reward-gradient backpropagation overhead. RWTD remains gradient-free and incurs only a modest overhead over DrPO, suggesting that frozen feature encoding and Sinkhorn transport contribute relatively little to the total training cost at the particle sizes used in our experiments. Overall, RWTD provides its alignment improvements at a computational cost comparable to existing methods, while remaining applicable to non-differentiable rewards.

B.3 Datasets and Models

In this section, we discuss the details of the datasets, reward models, and feature encoders used during training.

Experimental Breadth We evaluate RWTD across several orthogonal aspects of the alignment setting: two distinct large-scale one-step text-to-image generators (SANA Sprint 1.6B and SDXL-Turbo), two different frozen feature representations for transport (DINOv2 and MAE), and two qualitatively different post-training rewards, including non-differentiable compositional feedback and learned human-preference. The empirical results demonstrate that RWTD can generalize well across distinct generator and feature encoder backbones.

GenEval Training Prompts We post-train SDXL Turbo and SANA Sprint 1.6B using 800 total GenEval prompts, generated using the official GenEval codebase (MIT License) (Ghosh et al., 2023). We filter out prompts that overlap with the evaluation set to ensure fair evaluation. For SDXL Turbo training, the prompt set consists of 160 color attribution, 160 color, 160 counting, 160 position, and 160 two-object prompts. For SANA Sprint 1.6B training, the prompt set consists of 280 color attribution, 80 color, 160 counting, 140 position, 140 two-object prompts.

Pick-a-pic v2 In the preference alignment experiments, we post-train using the Pick-a-pic v2 prompt set (Kirstain et al., 2023). We use a filtered SFW subset of 15485 safe-for-work, user-written text prompts drawn from the Pick-a-Pic image-preference dataset. It contains diverse, naturally occurring text-to-image requests and excludes prompts flagged as explicit or otherwise unsafe.

GenEval training reward. Following prior work such as Liu et al. (2026); Zheng et al. (2026), which also uses dense GenEval-derived feedback for training, we use a dense adaptation of GenEval rather than the binary evaluation signal alone. Each generated image is evaluated against structured prompt metadata specifying the required object classes, counts, colors, and spatial relations, as well as any forbidden extra objects. A Mask2Former detector identifies COCO objects, while an OpenCLIP classifier predicts the color of each detected object crop. Following the official GenEval protocol, an image is marked correct only if all required objects and attributes are present, no forbidden count is reached, and every spatial relation is satisfied.

To provide a more fine-grained learning signal than binary correctness alone, we compute partial-credit scores for each metadata clause. Object-presence credit is the mean confidence of the top required detections, with missing detections assigned zero credit. Absence credit is one minus the confidence of the first forbidden extra detection. Color credit is the OpenCLIP probability assigned to the requested color, and spatial credit varies continuously with the normalized directional margin between object centers. The dense score is the mean over all applicable clauses,

$$r_{\text{dense}}(\boldsymbol{x}, c) = \frac{1}{|\mathcal{C}(c)|} \sum_{j \in \mathcal{C}(c)} s_j(\boldsymbol{x}, c), \qquad s_j \in [0, 1],$$

where $\boldsymbol{x}$ is the generated image, c is its structured prompt metadata, and $\mathcal{C}(c)$ denotes its presence, absence, color, and spatial-relation clauses. Our hybrid training reward additionally gives a bonus for satisfying the complete official GenEval criterion:

$$r_{\text{GenEval}}(\boldsymbol{x}, c) = r_{\text{dense}}(\boldsymbol{x}, c) + k\, \mathbf{1}[\text{GenEvalCorrect}(\boldsymbol{x}, c)], \qquad k = 0.50.$$

Thus, the dense component rewards incremental improvements toward satisfying individual constraints, while the binary bonus preserves the original benchmark objective of complete compositional correctness. The scorer is non-differentiable and is evaluated outside the model's computation graph. For fair comparison, we additionally trained all baselines with this dense reward.

HPSv2 Training Reward We use the official HPSv2 (Human Preference Score v2) reward model checkpoint (Wu et al., 2023). HPSv2 is a CLIP-style encoder trained from large-scale human judgments to score how well an image matches a prompt while reflecting human preferences for visual quality and aesthetics; higher scores indicate greater predicted human preference.

Feature Encoders RWTD and DrPO training rely on the use of a frozen feature encoder, following standard practices from previous work (Deng et al., 2026; Han et al., 2026a; Feng et al., 2026). For SDXL Turbo post-training, we use the MAE (He et al., 2022) encoder already used by the official DrPO codebase. For SANA Sprint 1.6B post-training, we use the DINOv2 (Oquab et al., 2023) encoder. Strong RWTD empirical results suggest that the objective can generalize well across both different one-step generator backbones and feature encoders.

(a) Reward temperature β

β	PS	HPSv2	CLIP	Aes.	IR
1	22.98	31.52	**0.2769**	6.696	1.289
2	23.02	31.92	0.2768	6.751	1.322
5	**23.04**	32.45	0.2761	6.800	1.358
10	23.00	**32.69**	0.2749	**6.824**	**1.366**

(b) Reference mass ρ

ρ	PS	HPSv2	CLIP	Aes.	IR
0	21.57	28.50	0.2561	**6.814**	1.071
0.15	**23.04**	**32.45**	0.2761	6.800	**1.358**
0.30	23.03	32.05	0.2766	6.713	1.325
0.45	23.00	31.74	**0.2772**	6.719	1.296
0.60	23.00	31.72	0.2771	6.687	1.294
0.75	23.00	31.64	0.2769	6.685	1.287
1.0	22.98	31.52	0.2766	6.681	1.277

(c) Partial transport step η

η	PS	HPSv2	CLIP	Aes.	IR
0.2	**23.04**	32.45	**0.2761**	6.800	**1.358**
0.4	23.03	32.48	0.2756	6.817	1.357
0.6	23.02	32.47	0.2754	**6.820**	1.353
0.8	23.01	**32.51**	0.2758	6.805	1.344
1.0	23.03	32.45	0.2758	6.783	1.353

Table 6: Ablations of the reward temperature β, reference mass ρ, and partial transport step size η for HPSv2 post-training on PartiPrompts. PS, Aes., and IR denote PickScore, Aesthetics, and ImageReward, respectively. Bold denotes the best result in each column.

B.4 Baseline Implementations

For the SDXL Turbo experiments, we used the official DrPO codebase and replaced its GenEval reward evaluator with our dense reward. This modification affected only reward computation; the DrPO objective and optimization procedure were otherwise unchanged. We trained DrPO for 800 optimizer steps, compared with 300 steps for RWTD, because shorter DrPO runs did not achieve competitive validation performance.

For SANA-Sprint-1.6B, no directly compatible implementations of DrPO and DRaFT were available, so we reimplemented them within the same training framework used for RWTD. For FAV, we used the official FAV codebase, adding our dense GenEval reward. All methods used the same pretrained SANA-Sprint-1.6B checkpoint, one-step generation configuration, and LoRA parameterization. We initialized method-specific hyperparameters from the corresponding papers or public implementations and performed tuning on held-out validation prompts. This tuning was restricted to method-specific quantities such as reward weighting, kernel temperatures, and update scales.

C Additional Results and Discussion

C.1 Ablation Studies

Hyperparameter Sweeps We perform coarse hyperparameter sweeps for the inverse reward temperature β, reference mass ρ, and partial transport step size η. Results are shown in Table 6.

Increasing the inverse reward temperature generally strengthens preference optimization: HPSv2, Aesthetics, and ImageReward improve as β increases. However, this sharper reward weighting is accompanied by a gradual decline in CLIP score, while PickScore peaks at $\beta = 5$. These results suggest a trade-off between aggressively concentrating on high-reward particles and preserving broader semantic alignment. We therefore use $\beta = 5$ as a balanced default.

While pure on-policy training ($\rho = 0$) results in severe degradation of most metrics, reflecting the overly aggressive distribution sharpening updates, we generally see that smaller (but non-zero) reference mass results in better optimization of the training reward HPSv2. Cross-reward generalization is demonstrated for all values $\rho > 0$, with $\rho = 0.15$ achieving the best PickScore, Aesthetics, and ImageReward, while $\rho = 0.45$ achieves the strongest CLIP score. We thus select $\rho = 0.15$ for its strong balance between in-domain reward optimization and cross-reward generalization.

Finally, we see that RWTD is relatively robust to the transport step size η, with PickScore, CLIP, and ImageReward peaking at $\eta = 0.2$, but with HPSv2 and Aesthetics peaking at larger η values. We choose $\eta = 0.2$ to balance cross-reward generalization with in-domain reward optimization.

Sample Diversity We additionally measure the diversity of images generated by our model by computing the average pairwise LPIPS (Zhang et al., 2018) distance between images per prompt. We use 200 prompts from the PartiPrompts set and generate 20 images per prompt, resulting in 190 pairwise comparisons per prompt.

Table 7 shows that purely on-policy RWTD ($\rho = 0$) substantially reduces perceptual diversity, reflecting the recursive reward tilting that concentrates the generator distribution. Introducing even modest reference mass ($\rho = 0.15$) increases average pairwise LPIPS from 0.414 to 0.580, and

Metric	SANA Sprint 1.6B	DRaFT	FAV	RWTD $\rho = 0$	RWTD $\rho = 0.15$	RWTD $\rho = 0.30$	RWTD $\rho = 0.45$	RWTD $\rho = 0.60$	RWTD $\rho = 0.75$	RWTD $\rho = 1.0$
Avg. pairwise LPIPS ↑	**0.652**	0.571	0.611	0.414	0.580	0.584	0.588	0.593	0.599	0.601

Table 7: Sample diversity measured by the average pairwise LPIPS distance between samples generated for the same prompt. Higher LPIPS indicates greater perceptual diversity.

Method	PickScore	HPSv2 (in-domain)	CLIP	Aesthetics	ImageReward
SANA Sprint 1.6B	22.75	30.31	0.2753	6.565	1.148
RWTD (final)	**23.04**	32.45	**0.2761**	6.800	**1.358**
DRaFT (step 100)	22.70	34.87	0.2655	7.089	1.291
DRaFT (step 400)	22.76	**37.02**	0.2573	**7.252**	1.312
FAV (step 100)	22.81	34.21	0.2711	6.844	1.281
FAV (step 400)	22.93	35.74	0.2718	6.949	1.322

Table 8: Early and final checkpoints of gradient-based baselines under HPSv2 post-training, evaluated on PartiPrompts. Step 100 is the earliest saved checkpoint for both baselines. Higher is better for all metrics; bold denotes the best value in each column.

the evaluated mixed configurations remain stable around 0.58–0.60. These values are in the same range as competing methods, despite RWTD's sharper mixed reward-tilted target. Although all aligned models remain below the base model (0.652), the results suggest that reference mixing substantially mitigates the diversity loss of pure on-policy tilting while retaining RWTD's reward-alignment benefits.

C.2 Does Early Stopping Benefit Gradient-based Baselines?

We evaluate the earliest saved checkpoints of DRaFT and FAV (step 100) to assess whether stopping earlier mitigates the semantic degradation observed after HPSv2 post-training. Table 8 compares these checkpoints with their final checkpoints, using the pretrained model and final RWTD model as references.

Both gradient-based baselines already exhibit reduced semantic alignment at step 100: HPSv2 increases substantially, while CLIP falls below the pretrained model. Earlier stopping partially mitigates DRaFT's CLIP degradation, whereas FAV's final checkpoint improves all reported metrics over its step-100 checkpoint. Neither early checkpoint recovers RWTD's combination of PickScore, CLIP, and ImageReward, despite achieving higher HPSv2. Thus, semantic degradation in these baselines is already present early in training, rather than arising only after prolonged optimization.

C.3 Discussion of Concurrent Work

Concurrent to our work, Hwang et al. (2026) propose reward-guided fine-tuning of one-step generators through Wasserstein gradient flow (WGF). Despite a shared optimal-transport perspective, the methods differ substantially in both their alignment objective and realization. Their approach formulates fine-tuning as a sequence of JKO steps toward the reward-weighted reference distribution and learns each step through a semi-dual adversarial objective. In contrast, RWTD defines an adaptive target that explicitly mixes separately reward-tilted current and frozen-reference distributions, computes a finite-sample feature-space OT coupling to this target, and amortizes the resulting assignments through direct feature regression. The concurrent work evaluates primarily on class-conditional image generation tasks, explicitly leaving larger scale text-to-image generation for future work; RWTD demonstrates effective alignment of such one-step text-to-image generators on compositional and preference objectives.

Table 9 provides a conceptual comparison of RWTD with recent alignment methods for one-step generative models.

Table 9: Conceptual comparison with recent methods for aligning one-step generative models. The key distinction of RWTD is its *model-dependent alignment target*: the current and reference distributions are separately reward-tilted and mixed, so the target itself evolves with the improving generator while retaining an explicit reference anchor.

Method	**Alignment object**	**Role of current model G_θ**	**Update realization**
FAV	Reward-tilted reference distribution	Provides samples to be moved toward a reference-defined target	Stein variational updates with kernel-density estimation
DrPO	Rank-based dipole preference field	Provides samples to rank to construct preferred feature-space directions	Regression toward detached feature targets induced by the preference field
WGF	Reward-tilted reference distribution	Current distribution is evolved toward the reward-tilted reference	Sequence of JKO steps learned through a semi-dual adversarial objective
RWTD	**Adaptive mixture of separately reward-tilted current and reference distributions**	**Current distribution explicitly enters the target, allowing newly discovered high-reward modes to affect subsequent updates**	**Feature-space optimal transport followed by direct fixed-point regression**

Algorithm 1 **Reward-Weighted Transport Distillation**

Input: Current generator G_θ, reference generator G_{ref}, reward r, frozen encoder ϕ
Input: Distributions p_z, p_{prompt}; parameters $\beta, \rho, \eta, \epsilon$; particle counts N, M
Input: Training steps T, Sinkhorn iterations L_{SK}, learning rate α

for $t = 1, \dots, T$ **do**
  Sample $c \sim p_{\mathrm{prompt}}$, $\{\boldsymbol{z}_i\}_{i=1}^N \sim p_z$, and $\{\boldsymbol{z}'_j\}_{j=1}^M \sim p_z$
  $\boldsymbol{x}_i \leftarrow G_\theta(\boldsymbol{z}_i, c), \quad i = 1, \dots, N$
  $\boldsymbol{y}_j \leftarrow G_{\mathrm{ref}}(\boldsymbol{z}'_j, c), \quad j = 1, \dots, M$
  $\boldsymbol{w}^\theta \leftarrow \mathrm{softmax}_i(\beta r(\boldsymbol{x}_i, c))$
  $\boldsymbol{w}^{\mathrm{ref}} \leftarrow \mathrm{softmax}_j(\beta r(\boldsymbol{y}_j, c))$
  $\boldsymbol{u}_i \leftarrow \phi(\boldsymbol{x}_i), \quad i = 1, \dots, N$
  $\{\boldsymbol{v}_k\}_{k=1}^{N+M} \leftarrow \{\boldsymbol{u}_i\}_{i=1}^N \,\|\, \{\phi(\boldsymbol{y}_j)\}_{j=1}^M$
  $\boldsymbol{a} \leftarrow \frac{1}{N}\mathbf{1}_N$
  $\boldsymbol{b} \leftarrow \left[(1-\rho)\boldsymbol{w}^\theta; \rho \boldsymbol{w}^{\mathrm{ref}}\right]$
  $\boldsymbol{C}_{ik} \leftarrow \|\boldsymbol{u}_i - \boldsymbol{v}_k\|_2^2$ for all i, k
  $\boldsymbol{P}_\epsilon \leftarrow \textsc{Sinkhorn}(\boldsymbol{C}, \boldsymbol{a}, \boldsymbol{b}, \epsilon, L_{\mathrm{SK}})$
  $\bar{\boldsymbol{u}}_i \leftarrow \dfrac{1}{a_i} \displaystyle\sum_{k=1}^{N+M} P_{\epsilon,ik} \boldsymbol{v}_k, \quad i = 1, \dots, N$
  $\widetilde{\boldsymbol{u}}_i \leftarrow (1-\eta)\boldsymbol{u}_i + \eta \bar{\boldsymbol{u}}_i, \quad i = 1, \dots, N$
  $\mathcal{L}_{\mathrm{RWTD}} \leftarrow \dfrac{1}{N} \displaystyle\sum_{i=1}^N \|\phi(G_\theta(\boldsymbol{z}_i, c)) - \mathrm{sg}(\widetilde{\boldsymbol{u}}_i)\|_2^2$
  $\theta \leftarrow \theta - \alpha \nabla_\theta \mathcal{L}_{\mathrm{RWTD}}$
end for
return θ

Algorithm 2 **Sinkhorn**

Input: Cost matrix $\boldsymbol{C} \in \mathbb{R}^{N \times K}$
Input: Marginals $\boldsymbol{a} \in \mathbb{R}_+^N$ and $\boldsymbol{b} \in \mathbb{R}_+^K$
Input: Regularization $\epsilon > 0$ and iterations L_{SK}

$\boldsymbol{K} \leftarrow \exp(-\boldsymbol{C}/\epsilon)$ ▷ Gibbs kernel
$\boldsymbol{t} \leftarrow \mathbf{1}_K$ ▷ Initialize
for $\ell = 1, \dots, L_{\mathrm{SK}}$ **do**
  $\boldsymbol{s} \leftarrow \boldsymbol{a} \oslash (\boldsymbol{K}\boldsymbol{t})$
  $\boldsymbol{t} \leftarrow \boldsymbol{b} \oslash (\boldsymbol{K}^\top \boldsymbol{s})$
end for
$\boldsymbol{P}_\epsilon \leftarrow \mathrm{diag}(\boldsymbol{s})\, \boldsymbol{K}\, \mathrm{diag}(\boldsymbol{t})$ ▷ Coupling
return $\boldsymbol{P}_\epsilon$

exp *and* $\oslash$ *are elementwise.*



(a) SANA Sprint 1.6B comparisons. RWTD improves alignment on difficult position prompts, while baselines omit the required second object or introduce visual artifacts.



(b) SDXL-Turbo comparisons. RWTD correctly satisfies the requested counts of four vases and three cows, and places the table to the right of the oven.

Figure 4: Qualitative GenEval results showing improved counting and spatial composition after post-training with RWTD.