RISE-DMD: REWARD-IMPROVED SELF-EVOLVING DISTRIBUTION MATCHING DISTILLATION FOR BLACK-BOX REWARD OPTIMIZATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Distribution Matching Distillation (DMD) can distill a multi-step diffusion model into a deterministic few-step generator, but introducing black-box rewards into DMD poses a key challenge: black-box rewards provide only scalar evaluations of final samples and do not directly specify the target distribution or marginal velocity field required by DMD. We propose RISE-DMD (Reward-Improved Self-Evolving Distribution Matching Distillation) and introduce Reward Score to learn a reward-improved marginal velocity field. Reward Score is trained on on-policy samples generated by the current student and their reward feedback, after which DMD uses the learned field to update the few-step student. The updated student then generates data for the next training round, allowing the target distribution to evolve together with the student distribution. Theoretically, we characterize the population-optimal Reward Score field and the corresponding reward-tilted target distributions at each diffusion time, and provide a regularized reward-optimization interpretation under a fixed-reference constraint. In the main SD3.5 Medium experiment, RISE-DMD outperforms the compared methods on all six preference and quality metrics, three of which are not used for training. Results on Sana 1.6B, DrawBench, and OCR further demonstrate the method's applicability across model backbones and prompt distributions, as well as its ability to optimize non-differentiable task rewards using only scalar feedback.

1 INTRODUCTION

Diffusion models generate high-quality samples, but iterative denoising requires multiple model evaluations. Distribution Matching Distillation (DMD) compresses a multi-step generative model into a deterministic few-step generator by minimizing the KL divergence between the forward-noised teacher and student marginals at each diffusion time and constructing the gradient from their marginal velocity fields. The student does not reproduce the full sampling trajectory (Yin et al., 2024b;a).

Although a few-step generator inherits the capabilities of its pretrained model, it is not directly optimized for specific downstream tasks. For example, the correctness of text in generated images can be evaluated using OCR outputs and edit distance, but such task evaluations typically provide only scalar feedback and no usable end-to-end gradient to the generator. We therefore seek to improve downstream task performance using only black-box rewards while retaining few-step inference efficiency. Direct reward-optimization methods such as DMDR rely on gradients from differentiable reward models and cannot directly handle this type of black-box feedback (Jiang et al., 2026).

RT-DM and the Re-DMD component of Reward Forcing incorporate black-box rewards by reweighting the original DMD gradient with terminal rewards (Jiang et al., 2026; Lu et al., 2026). Although motivated by reward-tilted target distributions, these updates still rely on the original teacher's marginal velocity field. Because they do not estimate the marginal velocity field of the reward-tilted target itself, the resulting reward-weighted gradient is generally not equivalent to the DMD gradient between the student and that target distribution. Moreover, as the student and teacher distributions become closer, the original DMD gradient becomes smaller, weakening the signal for the reward-weighted update.

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.

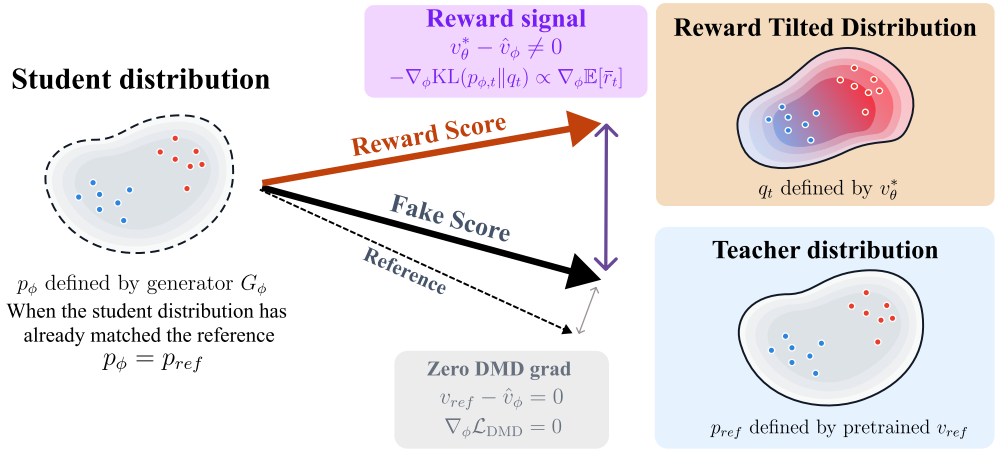

Figure 1: Matched-distribution special case. The student distribution has already matched the fixed teacher/reference distribution, $p_\phi = p_{\mathrm{ref}}$. Consequently, the Fake Score and reference fields coincide, $\hat{v}_\phi = v_{\mathrm{ref}}$, and the vanilla DMD gradient is zero. Reward Score still provides a reward signal and, through v_θ^*, defines a reward-tilted target distribution q_t distinct from the teacher distribution. The purple arrow indicates this target shift.

We address this problem by explicitly learning the marginal velocity field of the reward-tilted target distribution. We propose RISE-DMD, which formulates black-box reward optimization as an iterative process of reward-improved velocity-field learning and DMD student updates. We introduce an additional Reward Score model that learns the reward-improved marginal velocity field from the student's on-policy samples. The population-optimal Reward Score field corresponds to a family of reward-improved noisy target marginals, denoted by $q_{k,t}^R$. The subsequent DMD update matches the student's marginals to these reward-tilted targets. Because the updated student generates the on-policy samples for the next round, RISE-DMD continually matches a changing target distribution that evolves together with the student distribution.

The matched-distribution special case in Figure 1 makes this distinction explicit. When the teacher and student distributions already match, the original DMD gradient is zero, so a reward-weighted DMD gradient also remains zero. By contrast, the reward-improved marginal velocity field learned by Reward Score can still produce a nonzero DMD update that moves the student distribution toward higher-reward regions.

We evaluate RISE-DMD across model backbones and reward settings. Under a unified four-NFE SD3.5 Medium setting, it obtains the best result on all six evaluation metrics. Compared with DMD, HPSv2.1 improves from 0.323 to 0.353, while ImageReward, which was not used for training, improves from 1.297 to 1.519. Experiments across model backbones, test prompt distributions, and a non-differentiable OCR reward further demonstrate the applicability of the method.

Our contributions are:

- **Black-box reward optimization for DMD.** We propose RISE-DMD, which learns a reward-improved marginal velocity field from on-policy samples using scalar rewards and uses it as the DMD target for a few-step student.
- **Theoretical characterization.** We characterize the population-optimal Reward Score field, its reward-tilted marginals, and the regularized reward-optimization interpretation under a fixed reference constraint.
- **Unified empirical evaluation.** Under matched training and evaluation settings, RISE-DMD improves all six SD3.5 Medium metrics and remains effective across model backbones, prompt distributions, and a non-differentiable OCR reward.

2 Related Work

2.1 Few-Step Diffusion Distillation

Few-step diffusion distillation methods use different training signals. Trajectory and consistency methods learn coarse teacher transitions or enforce consistent predictions along sampling paths, as in Progressive Distillation (Salimans & Ho, 2022; Song et al., 2023; Luo et al., 2023; Zheng et al., 2024; Ren et al., 2024). Adversarial supervision can be added to distillation, as in Adversarial Diffusion Distillation (Sauer et al., 2024; Lin et al., 2024). Distribution matching instead minimizes a KL divergence between student and target noisy marginals without requiring paired teacher trajectories, as in DMD and its extensions (Yin et al., 2024b;a; Luo et al., 2025; Liu et al., 2026). Other one-step approaches reshape the data–noise coupling or use score-distillation supervision (Liu et al., 2023; 2024; Nguyen & Tran, 2024).

2.2 Reward Optimization for Diffusion and Flow Matching Models

Reward optimization for diffusion models and Flow Matching models can be organized by how feedback enters training. Differentiable-reward methods backpropagate through generated samples and therefore require a differentiable scorer (Xu et al., 2023; Prabhudesai et al., 2023). Reverse-process policy-gradient methods treat denoising transitions as stochastic actions and optimize scalar rewards with likelihood-ratio objectives and reference constraints (Black et al., 2024; Fan et al., 2023; Liu et al., 2025). Reward- and preference-weighted methods convert scalar scores or pairwise comparisons into weighted denoising or likelihood-based supervision, including objectives that avoid an explicit reward model (Lee et al., 2023; Wallace et al., 2024; Yang et al., 2024). Forward-process methods instead re-noise generated samples and learn from the resulting states, as in DiffusionNFT and CRD (Zheng et al., 2026; Zhu et al., 2026). These methods are designed around time-indexed diffusion or Flow Matching models, whereas the DMD student is a deterministic few-step generator, motivating reward-optimization methods tailored to few-step distillation.

2.3 Reward Optimization for Few-Step Generators

Existing approaches introduce rewards into few-step generation through direct reward gradients, reward-weighted distillation, or learned surrogates. Direct reward optimization can be combined with DMD as a distributional regularizer, as in DMDR's joint reward-and-DMD stage (Jiang et al., 2026). Reward-weighted distillation uses scalar rewards to rescale the existing DMD update; examples include the RT-DM branch of DMDR and the Re-DMD component of Reward Forcing (Jiang et al., 2026; Lu et al., 2026). Learned-surrogate approaches provide a differentiable signal from black-box feedback; TDM-R1 learns a diffusion-parameterized surrogate reward along deterministic generation trajectories (Luo et al., 2026).

3 Preliminaries

3.1 Flow Matching

Flow Matching (Lipman et al., 2023) trains a time-dependent velocity field by regressing to the conditional velocity of a prescribed probability path. In our notation, a clean sample x_0 is perturbed with Gaussian noise as

$$x_t = \alpha_t x_0 + \sigma_t \epsilon, \qquad \epsilon \sim \mathcal{N}(0, I), \qquad u_t = \dot{\alpha}_t x_0 + \dot{\sigma}_t \epsilon, \tag{1}$$

where u_t is the conditional path velocity. The corresponding marginal velocity field is $v_t(x_t) = \mathbb{E}[u_t \mid x_t]$. A velocity model v_ψ is fitted with the conditional Flow Matching objective

$$\mathcal{L}_{\mathrm{FM}}(\psi) = \mathbb{E}_{x_0,t,\epsilon}\left[\|v_\psi(x_t, t) - u_t\|_2^2\right]. \tag{2}$$

The Fake Score and Reward Score objectives below use this same velocity-field parameterization, with different targets or reward weights.

3.2 Distribution Matching Distillation with Velocity Fields

DMD trains a few-step generator by matching its output distribution to a target distribution (Yin et al., 2024b;a). Let $x_0 \sim p_\phi$ be a student sample and let p_{ref} be the fixed reference distribution. Under the shared forward noising process, let $p_{\phi,t}$ and $p_{\text{ref},t}$ denote the corresponding noisy marginals, with marginal velocity fields $v_{\phi,t}$ and $v_{\text{ref},t}$. Ignoring time-dependent weighting factors, the distribution-matching objective is

$$\mathcal{L}_{\text{DMD}}(\phi) = \mathbb{E}_t[D_{\text{KL}}(p_{\phi,t} \| p_{\text{ref},t})]\,. \tag{3}$$

After reparameterizing the noisy student state x_t, its gradient has the schematic form

$$\nabla_\phi \mathcal{L}_{\text{DMD}} \simeq \mathbb{E}\left[\left(\frac{\partial x_t}{\partial \phi}\right)^{\mathsf{T}} \left(\nabla_{x_t} \log p_{\phi,t}(x_t) - \nabla_{x_t} \log p_{\text{ref},t}(x_t)\right)\right]. \tag{4}$$

For a shared Gaussian forward process, the difference between these two log-density gradients is proportional to the marginal velocity difference,

$$\nabla_{x_t} \log p_{\phi,t} - \nabla_{x_t} \log p_{\text{ref},t} \propto v_{\phi,t} - v_{\text{ref},t}. \tag{5}$$

Thus, DMD can estimate the score-gradient direction from the difference between the target and student marginal velocity fields. In vanilla DMD, the pretrained teacher supplies v_{ref}. An online Fake Score model $\hat{v}_\phi$ is fitted to student-generated samples to estimate the marginal velocity field of the current student distribution.

3.3 DiffusionNFT Positive–Negative Flow Matching

DiffusionNFT (Zheng et al., 2026) learns a reward-improved marginal velocity field from generated samples and scalar rewards. It re-noises the samples through the forward process and trains velocity fields by Flow Matching, so data collection does not require reward gradients, denoising trajectories, or transition likelihoods. For the current student distribution p_ϕ, a normalized reward $r(x_0) \in [0,1]$ implicitly defines positive and negative distributions p_ϕ^+ and p_ϕ^-. Specifically,

$$p_\phi^+(x_0) \propto r(x_0) p_\phi(x_0), \qquad p_\phi^-(x_0) \propto (1 - r(x_0)) p_\phi(x_0). \tag{6}$$

Let v^+, v^-, and v_ϕ denote the marginal velocity fields of p_ϕ^+, p_ϕ^-, and p_ϕ, respectively. DiffusionNFT shows that $v^+ - v_\phi$ and $v_\phi - v^-$ are collinear. This relation allows the two branches to be parameterized by a single trainable velocity field v_θ:

$$v_\theta^+ = v_\phi + \beta(v_\theta - v_\phi), \qquad v_\theta^- = v_\phi - \beta(v_\theta - v_\phi). \tag{7}$$

The two branches are fitted by Flow Matching on the re-noised samples, weighted by r and $1 - r$, respectively. Up to normalization constants, this is equivalent to training on the positive and negative distributions. In RISE-DMD, the few-step student provides fresh on-policy samples and rewards each round, allowing Reward Score to learn a reward-improved field relative to the current student distribution. The practical center field and complete objective are given in Section 4.

4 RISE-DMD

4.1 Overview

Each round first draws samples from the four-step student, then learns a reward-improved marginal velocity field, and finally updates the student with DMD. Let $p_{\phi_k}(x_0)$ denote the student sampling distribution and $v_{\theta_k}(x_t, t)$ the reward-improved marginal velocity field learned from the on-policy samples in the current k^{th} iteration. The loop is

$$p_{\phi_k} \xrightarrow{\text{Reward Score learning}} v_{\theta_k} \xrightarrow{\text{DMD update}} p_{\phi_{k+1}}. \tag{8}$$

The updated student generates samples for the next round, so the learned marginal velocity field and the student distribution evolve together as summarized in eq. (8). The data flow is shown in fig. 2; the notation summary and full training algorithm are provided in Appendix A.2 and Appendix A.3, whose pseudocode is given in Algorithm 1.

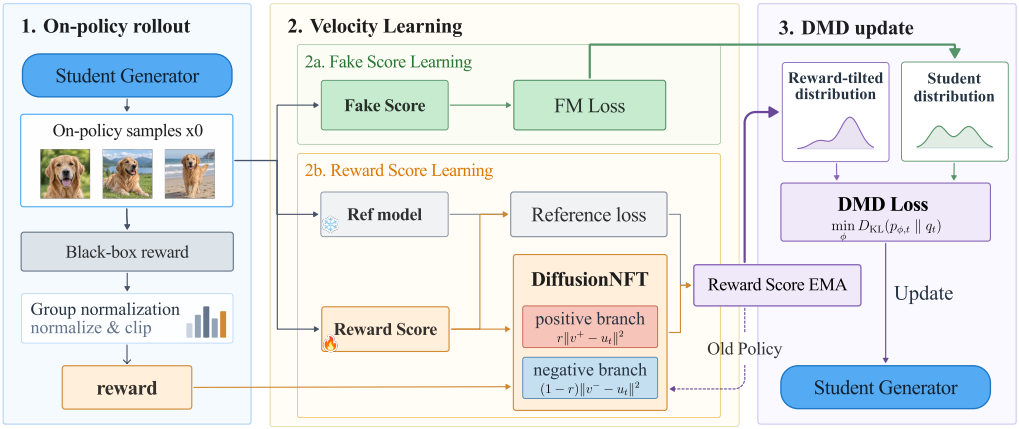

Figure 2: One RISE-DMD training round. (1) The four-step student generates and scores an on-policy buffer, whose rewards weight the positive and negative branches. (2a) Fake Score fits the current student field with eq. (9). (2b) Reward Score constructs the two branch fields in eq. (10), optimizes eq. (11), and updates its EMA. (3) The updated EMA provides the target-side velocity estimate, while Fake Score estimates the current student field for the DMD update in eq. (12).

4.2 Three-Stage Training Procedure

As shown in fig. 2, each round consists of on-policy sampling, marginal velocity-field learning, and a DMD student update. Stage 1 generates and scores samples. Stage 2 updates Fake Score and Reward Score in sequence. Stage 3 freezes both field models and updates only the four-step student.

Stage 1: on-policy sampling and reward construction. The current four-step student generates a sample buffer $x_0 \sim p_\phi$. A black-box evaluator returns a scalar reward for each generated sample. Normalization and clipping produce $r(x_0) \in [0, 1]$. We apply Gaussian forward noising to obtain x_t, which is used for field learning. The reward r and its complement $1 - r$ directly weight the positive and negative Flow Matching branches, respectively. We do not resample from separate positive and negative distributions; both weighted terms are evaluated on the same on-policy samples.

Stage 2: marginal velocity-field learning. **Stage 2a: Fake Score fitting.** We first train the Fake Score $\hat{v}_\phi$ on detached buffer samples to estimate the marginal velocity field v_ϕ of the current student distribution p_ϕ using eq. (9):

$$\mathcal{L}_{\text{fake}} = \mathbb{E}_{c,t,\epsilon,x_0\sim p_\phi(\cdot|c)} \left[\|\hat{v}_\phi(x_t, t) - u_t\|_2^2 \right]. \tag{9}$$

This update optimizes the Fake Score parameters; the student is not updated and $\hat{v}_\phi$ is frozen in the next stage.

Stage 2b: Reward Score training. Although the ideal construction in Section 3.3 centers the branches at the current student field v_ϕ, RISE-DMD uses the detached pre-update EMA field $v_{\bar{\theta}}$ as a stable old-field proxy. The Fake Score remains reserved for the current-student estimate in Stage 3, while $v_{\bar{\theta}}$ is fixed during this update. The positive and negative branch fields are constructed from the same online Reward Score in eq. (10):

$$v_\theta^+ = v_{\bar{\theta}} + \beta(v_\theta - v_{\bar{\theta}}), \qquad v_\theta^- = v_{\bar{\theta}} - \beta(v_\theta - v_{\bar{\theta}}). \tag{10}$$

Reward Score is optimized by the positive–negative Flow Matching objective together with one fixed-reference regularizer, as shown in eq. (11):

$$\mathcal{L}_{\text{RS}}(\theta) = \mathbb{E}_{c,t,\epsilon,x_0\sim p_\phi(\cdot|c)} \left[\frac{r}{\beta} \left\| v_\theta^+ - u_t \right\|_2^2 + \frac{1-r}{\beta} \left\| v_\theta^- - u_t \right\|_2^2 + \lambda_{\text{ref}} \left\| v_\theta - \text{sg}(v_{\text{ref}}) \right\|_2^2 \right]. \tag{11}$$

The field v_{ref} is the frozen reference model (fig. 2). This objective updates only θ; both $v_{\bar{\theta}}$ and v_{ref} are detached. Immediately after the Reward Score optimizer step, we update the EMA parameters and use the resulting field $v_{\bar{\theta}}$ for the subsequent student update.

Stage 3: DMD student update. At the DMD query state x_t, $\hat{v}_\phi$ provides the current student field and the updated $v_{\bar{\theta}}$ provides the target-side velocity estimate. We freeze all field models and update only the four-step student using eq. (12):

$$\mathcal{L}_{\text{student}}(\phi) = \mathcal{L}_{\text{DMD}}(v_{\bar{\theta}}, \hat{v}_\phi). \tag{12}$$

The online Reward Score and frozen reference do not define additional student losses. Because the DMD step is only an approximate projection, the next round reuses the resulting EMA as the detached old-field proxy while the updated student supplies fresh on-policy samples.

4.3 Population Optimum of Reward Score without Reference Regularization

To characterize the reward-dependent component of the objective, we set $\lambda_{\text{ref}} = 0$. Here r denotes the normalized and clipped reward used by DiffusionNFT. For a fixed noisy state (x_t, t), we hold $v_{\bar{\theta}}$ fixed. Expanding the positive–negative objective given in eq. (11) and minimizing the resulting quadratic objective gives

$$v^*_\theta(x_t, t) = v_{\bar{\theta}}(x_t, t) + \frac{1}{\beta}\mathbb{E}[(2r-1)(u_t - v_{\bar{\theta}}) \mid x_t]\,. \tag{13}$$

Using the Gaussian posterior identity, the optimum in eq. (13) is written as the reward correction:

$$v^*_\theta - v_{\bar{\theta}} = \frac{2\xi_t}{\beta}\nabla_{x_t}\bar{r}_t(x_t), \qquad \bar{r}_t(x_t) = \mathbb{E}[r(x_0) \mid x_t]. \tag{14}$$

Here ξ_t depends only on the Gaussian noise schedule. Thus, Reward Score adds a reward-dependent correction to the old velocity field. The detailed algebra is provided in Appendix A.4 and A.7.

4.4 Reward-Tilted Marginal Distribution and DMD Student Update

In the ideal analysis, let $v_{\bar{\theta}}$ be the marginal velocity field of the current student distribution $p_{\phi,t}$; in practice, it is estimated by the pre-update reward-improved target EMA. Let q_t denote the noisy marginal of the reward-improved target at diffusion time t. The velocity–score relation gives

$$v^*_\theta - v_{\bar{\theta}} = \xi_t \nabla_{x_t} \log \frac{q_t}{p_{\phi,t}}. \tag{15}$$

Combining eq. (15) with the optimum from Section 4.3 and integrating with respect to x_t yields

$$q_t(x_t) \propto p_{\phi,t}(x_t) \exp\left(\frac{2}{\beta}\bar{r}_t(x_t)\right). \tag{16}$$

Thus, Reward Score learns the marginal velocity field of the reward-tilted marginal q_t in eq. (16), which changes with the current student distribution.

For a local student update, we freeze q_t after constructing it from the current rollout. The gradient of its KL divergence at the current student distribution is given in eq. (17):

$$\nabla_\phi D_{\text{KL}}(p_{\phi,t} \| q_t) = -\frac{2}{\beta}\nabla_\phi \mathbb{E}_{p_{\phi,t}}[\bar{r}_t]. \tag{17}$$

Under the population analysis, with the constructed target q_t held fixed during the local student update, the marginal-velocity difference used by DMD corresponds to ascent on $\mathbb{E}_{p_{\phi,t}}[\bar{r}_t]$ (Appendix A.1). Thus, the update is not merely a sample-wise reweighting of the original DMD gradient. DiffusionNFT provides the positive–negative Flow Matching construction for learning the reward-improved marginal velocity field, while RISE-DMD uses the velocity–score relation to identify the associated reward-tilted marginal and applies DMD to match the student to this target.

4.5 Reference-Regularized Target and RLHF-Style Interpretation

We further discuss the case when $\lambda_{\text{ref}} > 0$. The reference regularizer changes the optimization target. Because the Reward Score objective is quadratic, its population-optimal velocity field is given in eq. (18):

$$v^*_{\text{reg}} = \frac{\beta v^*_\theta + \lambda_{\text{ref}} v_{\text{ref}}}{\beta + \lambda_{\text{ref}}} = \rho v^*_\theta + (1-\rho) v_{\text{ref}}, \qquad \rho = \frac{\beta}{\beta + \lambda_{\text{ref}}}. \tag{18}$$

Table 1: Unified comparison on SD3.5 Medium under a four-NFE setting.

Method	PickScore ↑	CLIPScore ↑	HPSv2.1 ↑	Aesthetic ↑	ImageReward ↑	UnifiedReward ↑
DMD	21.919	0.283	0.323	6.089	1.297	0.694
DMD2	21.965	0.285	0.326	6.157	1.318	0.695
DMDR	<u>22.409</u>	<u>0.287</u>	<u>0.339</u>	6.248	1.361	<u>0.710</u>
RT-DM	21.772	0.284	0.317	5.981	1.280	0.689
TDM-R1	22.398	<u>0.287</u>	0.338	<u>6.422</u>	<u>1.438</u>	0.696
RISE-DMD (ours)	**22.990**	**0.292**	**0.353**	**6.521**	**1.519**	**0.727**

Table 2: Sana 1.6B results under a four-NFE setting.

Method	PickScore ↑	CLIPScore ↑	HPSv2.1 ↑	Aesthetic ↑	ImageReward ↑	UnifiedReward ↑
DMD	21.624	<u>0.279</u>	0.314	<u>6.331</u>	1.216	0.688
RT-DM	<u>21.649</u>	**0.280**	<u>0.316</u>	6.319	<u>1.244</u>	<u>0.691</u>
RISE-DMD (ours)	**22.244**	0.275	**0.331**	**6.576**	**1.269**	**0.699**

Let $p_{\mathrm{ref},t}$ denote the noisy marginal of the frozen reference model under the same forward process. Using the velocity–score relation and the target from Section 4.4 gives the regularized target:

$$\begin{aligned} q_t^{\mathrm{reg}}(x_t) &\propto q_t(x_t)^{\rho} p_{\mathrm{ref},t}(x_t)^{1-\rho} \\ &\propto p_{\phi,t}(x_t)^{\rho} p_{\mathrm{ref},t}(x_t)^{1-\rho} \exp\left(\frac{2\bar{r}_t(x_t)}{\beta+\lambda_{\mathrm{ref}}}\right). \end{aligned} \tag{19}$$

When $\lambda_{\mathrm{ref}} = 0$, we have $\rho = 1$ and recover the target q_t from Section 4.4. For $\lambda_{\mathrm{ref}} > 0$, the reference term is therefore not redundant: it changes both the target distribution and the effective reward coefficient. For a local update, we freeze q_t^{reg}. Its gradient at the student distribution is:

$$\nabla_\phi D_{\mathrm{KL}}(p_{\phi,t} \| q_t^{\mathrm{reg}}) = \frac{\lambda_{\mathrm{ref}}}{\beta+\lambda_{\mathrm{ref}}} \nabla_\phi D_{\mathrm{KL}}(p_{\phi,t} \| p_{\mathrm{ref},t}) - \frac{2}{\beta+\lambda_{\mathrm{ref}}} \nabla_\phi \mathbb{E}_{p_{\phi,t}}[\bar{r}_t]. \tag{20}$$

Gradient descent on this KL combines reference regularization with reward ascent. Reference regularization changes the reward coefficient from $2/\beta$ to $2/(\beta+\lambda_{\mathrm{ref}})$. RISE-DMD learns the velocity field corresponding to this target and uses DMD to match the student to it; it does not backpropagate the black-box reward. The full derivation is provided in Appendix A.4.

5 Experiments

5.1 Experimental Setup and Compared Methods

All methods share the DMD initialization, SD3.5 Medium backbone (Esser et al., 2024), prompt stream, four-NFE sampler, and 900-update budget. RISE-DMD uses equal-weight PickScore, CLIPScore, and HPSv2.1 rewards; Aesthetic, ImageReward, and UnifiedReward are held out for evaluation. Each update processes 48 prompt groups of 24 samples, with matched prompt order and evaluation noise. We evaluate 2,048 Pick-a-Pic prompts with six metrics; detailed reward-processing and batching settings are given in Appendix A.6. We compare DMD, DMD2, RT-DM, DMDR, TDM-R1, and RISE-DMD.

5.2 Main and Generalization Results

RISE-DMD achieves the best result on all six metrics in table 1. Its gains extend to the three metrics not included in the training reward: Aesthetic (LAION, 2022), ImageReward (Xu et al., 2023), and UnifiedReward (Wang et al., 2025).

Figure 3 shows a matched-input comparison of DMD, RT-DM, and RISE-DMD; additional examples are provided in Appendix B.

We further test transfer across model backbones by applying the same comparison to Sana 1.6B (Xie et al., 2025) at 1024 pixels. RISE-DMD leads on five of six metrics, supporting transfer across architectures and resolutions. Results are reported in Table 2.

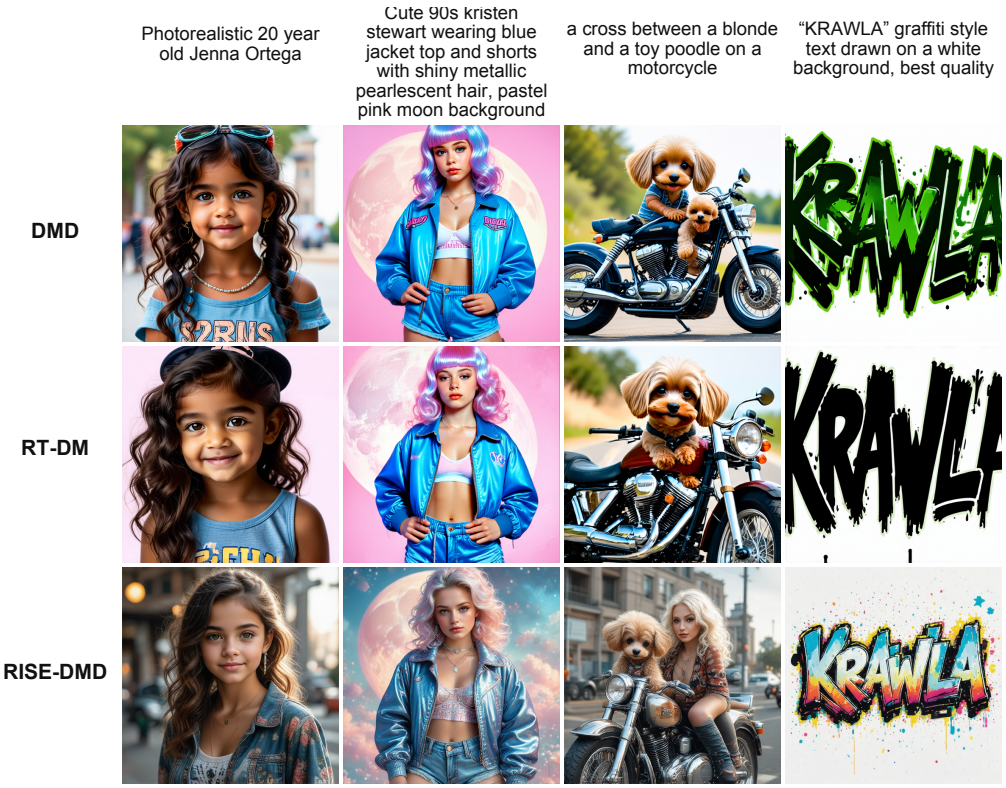

Figure 3: Matched-input comparison of DMD, RT-DM, and RISE-DMD at four NFE.

Table 3: Automatic DrawBench evaluation with four NFE.

Method	PickScore ↑	CLIPScore ↑	HPSv2.1 ↑	Aesthetic ↑	ImageReward ↑	UnifiedReward ↑
DMD	22.576	0.283	0.310	5.546	1.164	0.649
RT-DM	22.549	0.285	0.306	5.489	1.196	0.649
RISE-DMD (ours)	**23.477**	**0.287**	**0.331**	**5.984**	**1.380**	**0.688**

We also evaluate generalization beyond the training prompt distribution on DrawBench (Saharia et al., 2022). We use the `test.txt` file released with DiffusionNFT (Zheng et al., 2026), which contains 200 unique prompts repeated five times. All methods share the prompt order, initial noise, base random seed, SD3.5 Medium backbone, and four-NFE sampling configuration. Results are reported in table 3.

RISE-DMD obtains the best result on all six metrics, with consistent gains over DMD and RT-DM.

5.3 Reward-Field Ablations and Black-Box Reward Optimization

We also compare reward normalization and clipping under the same training protocol. Group std with clip $= 1.0$ improves HPSv2.1, ImageReward, and UnifiedReward and is used in the main experiment; the full comparison is reported in Appendix A.5.

We first examine the choice of the previous-field estimate. In this controlled 200-update record, using the Reward Score EMA as $v_{\bar{\theta}}$ gives higher online reward and a larger Reward-DMD field change. Figure 4 shows all 200 updates. The EMA provides a gradually changing reference for learning the reward-driven velocity correction.

We next compare the positive–negative objective with linear and exponential reward-weighted Flow Matching. Over 200 updates, both weighted variants decrease from an online reward of about 1.430 to 0.82, whereas RISE-DMD reaches 1.511 (fig. 5). This result supports the positive–negative ob-

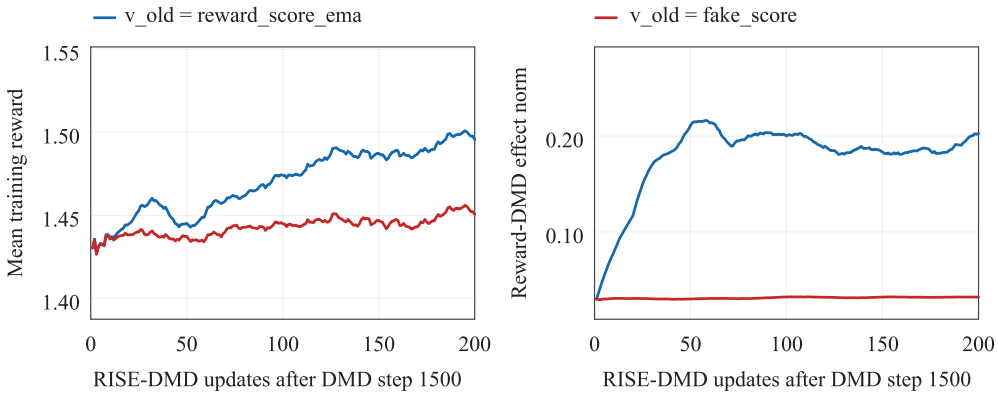

Figure 4: Previous-field source ablation over the first 200 updates.

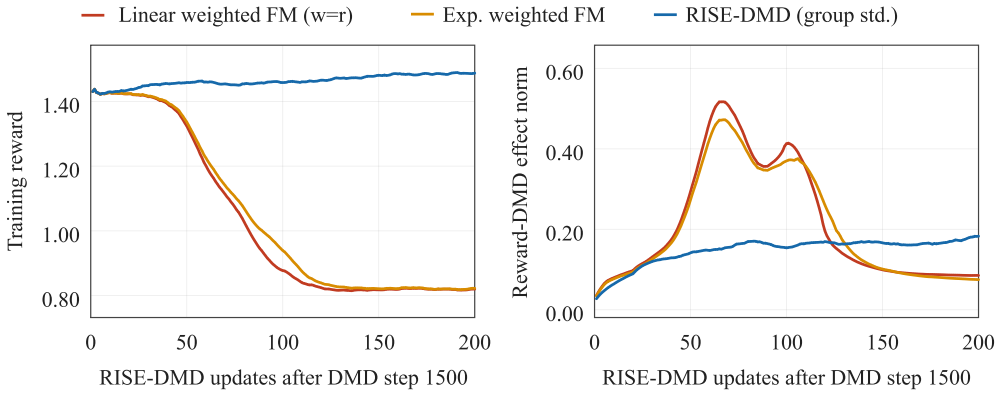

Figure 5: Reward-field learning objective ablation over the first 200 on-policy updates.

jective as the more effective reward-field learning signal in the present on-policy few-step setting; matched-input examples are provided in Appendix B.

Finally, we test optimization of a non-differentiable OCR reward. We compute the OCR reward using PaddleOCR (Du et al., 2020) and normalized edit distance, combine it with PickScore, CLIP-Score, and HPSv2.1 using equal weights, and use only scalar outputs during training. On 1,018 fixed prompts, the OCR score increases from 0.412 for DMD to 0.889 after 300 updates with Type-1 EMA. The online reward curves for both EMA schedules are provided in Appendix C.

6 Discussion and Conclusion

RISE-DMD converts terminal scalar rewards into a per-round reward-tilted target that a deterministic four-step student can match. Experiments on SD3.5 Medium, Sana 1.6B, held-out prompts, and a non-differentiable OCR task demonstrate improved few-step generation and black-box reward optimization using scalar feedback only. Our analysis characterizes the population objective under a squared loss, an ideal old field, and a fixed per-round target; it does not establish global convergence of the practical neural training procedure or monotonic improvement of the terminal reward.

AI USE STATEMENT

We used generative AI tools to aid manuscript prose and to check LaTeX organization, citations, and cross-references. We did not use generative AI tools to generate experimental data, numerical results, or scientific claims; other required-disclosure categories are not applicable to this work. The authors reviewed these outputs against the manuscript and source records, verified the citations and references, and take responsibility for the final manuscript.

REPRODUCIBILITY STATEMENT

The main text defines the RISE-DMD objective and update order. The appendix provides the notation, training pseudocode, closed-form derivations, Gaussian forward-process identity, reward normalization settings, and matched evaluation protocols. The reported comparisons use shared prompts, fixed seeds, and the same four-NFE sampler within each study. Code, training configurations, and evaluation scripts will be released upon acceptance.

REFERENCES

Kevin Black, Michael Janner, Yilun Du, Ilya Kostrikov, and Sergey Levine. Training diffusion models with reinforcement learning. In *International Conference on Learning Representations*, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/hash/14f75513f0f1ca01de1e826b52e6b840-Abstract-Conference.html.

Yuning Du, Chenxia Li, Ruoyu Guo, Xiaoting Yin, Weiwei Liu, Jun Zhou, Yifan Bai, Zilin Yu, Yehua Yang, Qingqing Dang, and Haoshuang Wang. Pp-ocr: A practical ultra lightweight ocr system. *arXiv preprint arXiv:2009.09941*, 2020. URL https://arxiv.org/abs/2009.09941.

Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, Dustin Podell, Tim Dockhorn, Zion English, and Robin Rombach. Scaling rectified flow transformers for high-resolution image synthesis. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235, pp. 12606–12633, 2024. URL https://proceedings.mlr.press/v235/esser24a.html.

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. In *Advances in Neural Information Processing Systems*, volume 36, 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/fc65fab891d83433bd3c8d966edde311-Abstract.html.

Jack Hessel, Ari Holtzman, Maxwell Forbes, Ronan Le Bras, and Yejin Choi. Clipscore: A reference-free evaluation metric for image captioning. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pp. 7514–7528, 2021. URL https://aclanthology.org/2021.emnlp-main.595/.

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

Dengyang Jiang, Dongyang Liu, Zanyi Wang, Qilong Wu, Liuzhuozheng Li, Hengzhuang Li, Xin Jin, Changsheng Lu, Zhen Li, Mengmeng Wang, Steven Hoi, Peng Gao, and Harry Yang. Distribution matching distillation meets reinforcement learning. In *European Conference on Computer Vision*, pp. 281–299, 2026. doi: 10.1007/978-3-032-37550-6_16. URL https://link.springer.com/chapter/10.1007/978-3-032-37550-6_16.

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

LAION. Laion-aesthetics, 2022. URL https://laion.ai/blog/laion-aesthetics/.

Kimin Lee, Hao Liu, Moonkyung Ryu, Olivia Watkins, Yuqing Du, Craig Boutilier, Pieter Abbeel, Mohammad Ghavamzadeh, and Shixiang Shane Gu. Aligning text-to-image models using human feedback. *arXiv preprint arXiv:2302.12192*, 2023. URL https://arxiv.org/abs/2302.12192.

Shanchuan Lin, Anran Wang, and Xiao Yang. Sdxl-lightning: Progressive adversarial diffusion distillation. *arXiv preprint arXiv:2402.13929*, 2024. URL https://arxiv.org/abs/2402.13929.

Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matthew Le. Flow matching for generative modeling. In *International Conference on Learning Representations*, 2023. URL https://mlanthology.org/iclr/2023/lipman2023iclr-flow/.

Jie Liu, Gongye Liu, Jiajun Liang, Yangguang Li, Jiaheng Liu, Xintao Wang, Pengfei Wan, Di Zhang, and Wanli Ouyang. Flow-grpo: Training flow matching models via online rl. In *Advances in Neural Information Processing Systems*, volume 38, Main Conference, pp. 40783–40818, 2025. doi: 10.52202/085713-1362. URL https://proceedings.neurips.cc/paper_files/paper/2025/hash/3a10c46572628d58cb44fb705f25cbbf-Abstract-Conference.html.

Tao Liu, Hao Yan, Mengting Chen, Taihang Hu, Zhengrong Yue, Zihao Pan, Jinsong Lan, Xiaoyong Zhu, Ming-Ming Cheng, Bo Zheng, and Yaxing Wang. Continuous-time distribution matching for few-step diffusion distillation. *arXiv preprint arXiv:2605.06376*, 2026. URL https://arxiv.org/abs/2605.06376.

Xingchao Liu, Chengyue Gong, and Qiang Liu. Flow straight and fast: Learning to generate and transfer data with rectified flow. In *International Conference on Learning Representations*, 2023. URL https://mlanthology.org/iclr/2023/liu2023iclr-flow/.

Xingchao Liu, Xiwen Zhang, Jianzhu Ma, Jian Peng, and Qiang Liu. Instaflow: One step is enough for high-quality diffusion-based text-to-image generation. In *International Conference on Learning Representations*, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/hash/4dc37a7bc61057252ce043fa3b83aac2-Abstract-Conference.html.

Yunhong Lu, Yanhong Zeng, Haobo Li, Hao Ouyang, Qiuyu Wang, Ka Leong Cheng, Jiapeng Zhu, Hengyuan Cao, Zhipeng Zhang, Xing Zhu, Yujun Shen, and Min Zhang. Reward forcing: Efficient streaming video generation with rewarded distribution matching distillation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 34385–34397, 2026. URL https://openaccess.thecvf.com/content/CVPR2026/html/Lu_Reward_Forcing_Efficient_Streaming_Video_Generation_with_Rewarded_Distribution_Matching_CVPR_2026_paper.html.

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

Yihong Luo, Tianyang Hu, Jiacheng Sun, Yujun Cai, and Jing Tang. Learning few-step diffusion models by trajectory distribution matching. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 17719–17728, 2025. URL https://openaccess.thecvf.com/content/ICCV2025/html/Luo_Learning_Few-Step_Diffusion_Models_by_Trajectory_Distribution_Matching_ICCV_2025_paper.html.

Yihong Luo, Tianyang Hu, Weijian Luo, and Jing Tang. Tdm-r1: Reinforcing few-step diffusion models with non-differentiable reward. *arXiv preprint arXiv:2603.07700*, 2026. URL https://arxiv.org/abs/2603.07700.

Thuan Hoang Nguyen and Anh Tran. Swiftbrush: One-step text-to-image diffusion model with variational score distillation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 7807–7816, 2024. URL https:

//openaccess.thecvf.com/content/CVPR2024/html/Nguyen_SwiftBrush_One-Step_Text-to-Image_Diffusion_Model_with_Variational_Score_Distillation_CVPR_2024_paper.html.

Mihir Prabhudesai, Anirudh Goyal, Deepak Pathak, and Katerina Fragkiadaki. Aligning text-to-image diffusion models with reward backpropagation. *arXiv preprint arXiv:2310.03739*, 2023. URL https://arxiv.org/abs/2310.03739.

Yuxi Ren, Xin Xia, Yanzuo Lu, Jiacheng Zhang, Jie Wu, Pan Xie, Xing Wang, and Xuefeng Xiao. Hyper-sd: Trajectory segmented consistency model for efficient image synthesis. In *Advances in Neural Information Processing Systems*, volume 37, 2024. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/d4e1c24ac41ff0b82ca1b171731f0b23-Abstract-Conference.html.

Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily Denton, Seyed Kamyar Seyed Ghasemipour, Burcu Karagol Ayan, S. Sara Mahdavi, Rapha Gontijo Lopes, Tim Salimans, Jonathan Ho, David J. Fleet, and Mohammad Norouzi. Photorealistic text-to-image diffusion models with deep language understanding. In *Advances in Neural Information Processing Systems*, volume 35, 2022. URL https://proceedings.neurips.cc/paper_files/paper/2022/hash/ec795aeadae0b7d230fa35cbaf04c041-Abstract.html.

Tim Salimans and Jonathan Ho. Progressive distillation for fast sampling of diffusion models. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/pdf?id=TIdIXIpzhoI.

Axel Sauer, Dominik Lorenz, Andreas Blattmann, and Robin Rombach. Adversarial diffusion distillation. In *European Conference on Computer Vision*, 2024. URL https://www.ecva.net/papers/eccv_2024/papers_ECCV/html/11557_ECCV_2024_paper.php.

Yang Song, Prafulla Dhariwal, Mark Chen, and Ilya Sutskever. Consistency models. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202, pp. 32211–32252, 2023. URL https://proceedings.mlr.press/v202/song23a.html.

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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 8228–8238, 2024. URL https://openaccess.thecvf.com/content/CVPR2024/html/Wallace_Diffusion_Model_Alignment_Using_Direct_Preference_Optimization_CVPR_2024_paper.html.

Yibin Wang, Yuhang Zang, Hao Li, Cheng Jin, and Jiaqi Wang. Unified reward model for multimodal understanding and generation. *arXiv preprint arXiv:2503.05236*, 2025. URL https://arxiv.org/abs/2503.05236.

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

Enze Xie, Junsong Chen, Junyu Chen, Han Cai, Haotian Tang, Yujun Lin, Zhekai Zhang, Muyang Li, Ligeng Zhu, Yao Lu, and Song Han. Sana: Efficient high-resolution image synthesis with linear diffusion transformers. In *International Conference on Learning Representations*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/34345e243156da67605d4b63d71c8d98-Abstract-Conference.html.

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

Kai Yang, Jian Tao, Jiafei Lyu, Chunjiang Ge, Jiaxin Chen, Weihan Shen, Xiaolong Zhu, and Xiu Li. Using human feedback to fine-tune diffusion models without any reward model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 8941–8951, 2024. URL https://openaccess.thecvf.com/content/CVPR2024/html/Yang_Using_Human_Feedback_to_Fine-tune_Diffusion_Models_without_Any_Reward_CVPR_2024_paper.html.

Tianwei Yin, Michaël Gharbi, Taesung Park, Richard Zhang, Eli Shechtman, Frédo Durand, and William T. Freeman. Improved distribution matching distillation for fast image synthesis. In *Advances in Neural Information Processing Systems*, volume 37, 2024a. doi: 10.52202/079017-1505. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/54dcf25318f9de5a7a01f0a4125c541e-Abstract-Conference.html.

Tianwei Yin, Michaël Gharbi, Richard Zhang, Eli Shechtman, Frédo Durand, William T. Freeman, and Taesung Park. One-step diffusion with distribution matching distillation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 6613–6623, 2024b. doi: 10.1109/CVPR52733.2024.00632. URL https://openaccess.thecvf.com/content/CVPR2024/html/Yin_One-step_Diffusion_with_Distribution_Matching_Distillation_CVPR_2024_paper.html.

Jianbin Zheng, Minghui Hu, Zhongyi Fan, Chaoyue Wang, Changxing Ding, Dacheng Tao, and Tat-Jen Cham. Trajectory consistency distillation: Improved latent consistency distillation by semi-linear consistency function with trajectory mapping. *arXiv preprint arXiv:2402.19159*, 2024. URL https://arxiv.org/abs/2402.19159.

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*, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/hash/d8e68ddfe22520b45b8fb8d5cbde5a21-Abstract-Conference.html.

Yuanzhi Zhu, Xi Wang, Stéphane Lathuilière, and Vicky Kalogeiton. Diffusion reinforcement learning via centered reward distillation. *arXiv preprint arXiv:2603.14128*, 2026. URL https://arxiv.org/abs/2603.14128.

A Method Details and Supplementary Theory

A.1 Reward-Tilted Targets and Sample-Level Weighting

Reward-tilted distribution. Reward Forcing's Re-DMD (Lu et al., 2026) exponentially tilts the reference distribution, as shown in eq. (21):

$$\tilde{q}_R(x_0 \mid c) \propto p_{\text{ref}}(x_0 \mid c) \exp\left(\frac{r(x_0, c)}{\beta_R}\right), \tag{21}$$

where r is the terminal reward and β_R controls the tilting strength.

KL gradient for exact target matching. Directly minimizing the KL divergence between the student and reward-tilted distributions decomposes the gradient into the vanilla DMD KL gradient and an additional reward term, as shown in eq. (22):

$$\nabla_\phi \, \mathrm{KL}(p_\phi \| \tilde{q}_R) = \underbrace{\nabla_\phi \, \mathrm{KL}(p_\phi \| p_{\text{ref}})}_{\text{vanilla DMD KL gradient}} - \underbrace{\frac{1}{\beta_R} \nabla_\phi \mathbb{E}_{p_\phi}[r]}_{\text{additional reward-induced term}} . \tag{22}$$

Reward-weighted gradient used by Reward Forcing. Re-DMD instead uses

$$\mathbb{E}[\mathrm{sg}(w_R)\, g_{\text{DMD}}], \qquad w_R \propto \exp\left(\frac{r}{\beta_R}\right). \tag{23}$$

Table 4: Main notation used throughout the method and analysis.

Symbol	Meaning
G_ϕ	Four-step student generator with parameters ϕ.
p_ϕ, $p_{\phi,t}$	Clean-sample distribution generated by G_ϕ and the noisy marginal induced by the forward process at time t; at round k, p_{ϕ_k} and $p_{\phi_k,t}$.
x_0	Clean sample generated by the student.
x_t	Forward-noised sample at time t.
u_t	Conditional velocity of the forward path at x_t.
v_ϕ	Ideal marginal velocity field of the student distribution p_ϕ.
p_{ref}, $p_{\mathrm{ref},t}$, v_{ref}, $v_{\mathrm{ref},t}$	Fixed reference distribution, its noisy marginal at time t, and the corresponding marginal velocity fields.
$r(x_0)$	Normalized and clipped bounded reward weight.
θ, v_θ	Reward Score parameters and the full marginal velocity field produced by the model.
$\bar{\theta}$, $v_{\bar{\theta}}$	EMA parameters of Reward Score and the corresponding marginal velocity field; before the Reward Score update, $v_{\bar{\theta}}$ is the detached estimate of the current student's marginal field.
$\hat{v}_\phi$	Estimate of v_ϕ fitted to the student buffer.
v_θ^*	Population-optimal marginal velocity field of the DiffusionNFT objective without fixed-reference regularization.
v_{reg}^*	Population-optimal marginal velocity field after adding fixed-reference regularization.
q_t	Unregularized target marginal distribution corresponding to v_θ^*.
q_t^{reg}	Target marginal distribution after adding fixed-reference regularization.
ρ	Mixture coefficient between the current-student target and the fixed-reference target, $\beta/(\beta+\lambda_{\mathrm{ref}})$.
$\bar{r}_t(x_t)$	Conditional expected reward given x_t, $\mathbb{E}[r(x_0) \mid x_t]$.

This operation only rescales the vanilla DMD gradient in eq. (23) and does not explicitly add the reward-induced term in eq. (22). Reward tilting and sample-level reward weighting are therefore generally different. RISE-DMD addresses this distinction by learning the reward-improved marginal velocity field with Reward Score, thereby incorporating the reward-induced gradient direction indirectly without differentiating through the black-box reward.

A.2 Symbol Convention

Table 4 summarizes the main symbols used throughout the method and analysis. We omit the round index in single-round derivations and restore it when discussing the per-round target.

In the implementation, the fixed reference field is the frozen reference-model field. Coefficients used only in the analysis are defined when they first appear.

A.3 Training Algorithm

The following subsections provide the closed-form derivations omitted from the main text, the Gaussian forward-process identity, and an analysis of previous-field mismatch. Unless stated otherwise, we fix the prompt condition and omit it from the notation, and use the standard squared population Flow Matching objective. The practical branch normalization, finite optimization, and parameter EMA used in the main method are not assumptions of the analytical derivations below.

A.4 Supplementary Closed-Form Derivations

This section provides the algebra omitted from Sections 4.3–4.5. For fixed (x_t, t), substituting the positive and negative branches from Section 4.2 into the unregularized squared objective and collecting terms independent of v_θ into $C(x_t)$ gives

$$\mathcal{L}_{\mathrm{NFT}}(v_\theta) = C(x_t) + \beta\|v_\theta - v_{\bar{\theta}}\|_2^2 - 2(v_\theta - v_{\bar{\theta}})^{\mathsf{T}}\mathbb{E}[(2r-1)(u_t - v_{\bar{\theta}}) \mid x_t].$$

Algorithm 1 RISE-DMD training.

Require: Student G_ϕ, Reward Score parameters θ and EMA parameters $\bar{\theta}$, fake score $\hat{v}_\phi$, frozen reference v_{ref}, and black-box reward R

while training **do**
  Generate an on-policy student rollout buffer $\mathcal{B}$
  Evaluate R and map normalized within-prompt rewards to r
  Update $\hat{v}_\phi$ on $\mathcal{B}$ with Flow Matching
  Use the detached pre-update EMA field $v_{\bar{\theta}}$ as the current-student field estimate
  Construct v_θ^+ and v_θ^-; update θ with the DiffusionNFT-style loss
  Update the EMA parameters immediately after the Reward Score update
  Update G_ϕ with the DMD loss, using $v_{\bar{\theta}}$ as the target-side velocity estimate and $\hat{v}_\phi$ as the current-student field estimate
end while
return G_ϕ

Differentiating with respect to v_θ and setting the derivative to zero gives the optimum in Section 4.3:

$$v_\theta^* = v_{\bar{\theta}} + \frac{1}{\beta}\mathbb{E}[(2r-1)(u_t - v_{\bar{\theta}}) \mid x_t].$$

In the population analysis, $v_{\bar{\theta}}$ is the velocity field of the current on-policy marginal, so $\mathbb{E}[u_t - v_{\bar{\theta}} \mid x_t] = 0$ and

$$v_\theta^* = v_{\bar{\theta}} + \frac{2}{\beta}\mathbb{E}[r(x_0)(u_t - v_{\bar{\theta}}) \mid x_t].$$

Substituting the Gaussian posterior identity from Appendix A.7 and then applying the velocity–score relation gives the reward-tilted target in Section 4.4:

$$q_t(x_t) \propto p_{\phi,t}(x_t)\exp\left(\frac{2}{\beta}\bar{r}_t(x_t)\right).$$

We next add the fixed-reference term. Completing the square in the unregularized objective gives

$$\mathcal{L}_{\text{RS}}(v_\theta \mid x_t, t) = \beta\|v_\theta - v_\theta^*\|_2^2 + \lambda_{\text{ref}}\|v_\theta - v_{\text{ref}}\|_2^2 + C(x_t, t).$$

Its derivative with respect to v_θ is

$$2\beta(v_\theta - v_\theta^*) + 2\lambda_{\text{ref}}(v_\theta - v_{\text{ref}}).$$

Setting this derivative to zero and defining $\rho = \beta/(\beta + \lambda_{\text{ref}})$ gives

$$v_{\text{reg}}^* = \frac{\beta v_\theta^* + \lambda_{\text{ref}} v_{\text{ref}}}{\beta + \lambda_{\text{ref}}} = \rho v_\theta^* + (1-\rho)v_{\text{ref}}.$$

To identify the associated marginal, apply the velocity–score relation from Appendix A.7 to the two component fields:

$$v_\theta^* = v_{\bar{\theta}} + \xi_t \nabla_{x_t}\log\frac{q_t}{p_{\phi,t}}, \qquad v_{\text{ref}} = v_{\bar{\theta}} + \xi_t\nabla_{x_t}\log\frac{p_{\text{ref},t}}{p_{\phi,t}}.$$

Substitution into the convex combination yields

$$\begin{aligned} v_{\text{reg}}^* &= v_{\bar{\theta}} + \xi_t\nabla_{x_t}\left[\rho\log\frac{q_t}{p_{\phi,t}} + (1-\rho)\log\frac{p_{\text{ref},t}}{p_{\phi,t}}\right] \\ &= v_{\bar{\theta}} + \xi_t\nabla_{x_t}\log\frac{q_t^\rho p_{\text{ref},t}^{1-\rho}}{p_{\phi,t}}. \end{aligned}$$

Therefore, integration with respect to x_t, with the normalization factor absorbed into the proportionality constant, gives

$$q_t^{\text{reg}}(x_t) \propto q_t(x_t)^\rho p_{\text{ref},t}(x_t)^{1-\rho}.$$

Substituting $q_t \propto p_{\phi,t}\exp(2\bar{r}_t/\beta)$ and using $\rho/\beta = 1/(\beta + \lambda_{\text{ref}})$ gives

$$q_t^{\text{reg}}(x_t) \propto p_{\phi,t}(x_t)^\rho p_{\text{ref},t}(x_t)^{1-\rho}\exp\left(\frac{2}{\beta+\lambda_{\text{ref}}}\bar{r}_t(x_t)\right).$$

Table 5: Reward normalization and clipping configurations on SD3.5 Medium.

Configuration	PickScore ↑	CLIPScore ↑	HPSv2.1 ↑	Aesthetic ↑	ImageReward ↑	UnifiedReward ↑
Global std, clip $= 0.5$	**23.000**	**0.293**	0.342	**6.522**	1.485	0.724
Global std, clip $= 1.0$	22.624	**0.293**	0.327	6.382	1.437	0.706
Group std, clip $= 1.0$	22.990	0.292	**0.353**	6.521	**1.519**	**0.727**

Writing the normalization constant as Z_t, its log density is

$$\log q_t^{\mathrm{reg}} = \rho \log p_{\phi,t} + (1-\rho)\log p_{\mathrm{ref},t} + \frac{2}{\beta+\lambda_{\mathrm{ref}}}\bar{r}_t - \log Z_t.$$

Let $\tilde{p}_t$ be any normalized candidate marginal optimized in the current round. Substituting the expression above into $D_{\mathrm{KL}}(\tilde{p}_t \| q_t^{\mathrm{reg}}) = \mathbb{E}_{\tilde{p}_t}[\log \tilde{p}_t - \log q_t^{\mathrm{reg}}]$ gives

$$\begin{aligned} D_{\mathrm{KL}}(\tilde{p}_t \| q_t^{\mathrm{reg}}) = \rho D_{\mathrm{KL}}(\tilde{p}_t \| p_{\phi,t}) + (1-\rho) D_{\mathrm{KL}}(\tilde{p}_t \| p_{\mathrm{ref},t}) \\ - \frac{2}{\beta+\lambda_{\mathrm{ref}}} \mathbb{E}_{\tilde{p}_t}[\bar{r}_t] + \log Z_t. \end{aligned}$$

Multiplying by $(\beta+\lambda_{\mathrm{ref}})/2$ yields

$$\begin{aligned} \frac{\beta+\lambda_{\mathrm{ref}}}{2} D_{\mathrm{KL}}(\tilde{p}_t \| q_t^{\mathrm{reg}}) = \frac{\beta}{2} D_{\mathrm{KL}}(\tilde{p}_t \| p_{\phi,t}) + \frac{\lambda_{\mathrm{ref}}}{2} D_{\mathrm{KL}}(\tilde{p}_t \| p_{\mathrm{ref},t}) \\ - \mathbb{E}_{\tilde{p}_t}[\bar{r}_t] + \frac{\beta+\lambda_{\mathrm{ref}}}{2} \log Z_t. \end{aligned}$$

The last term is independent of $\tilde{p}_t$. Hence minimizing the left-hand side is equivalent to maximizing expected reward under KL penalties to the current student and the fixed reference, and the unique normalized optimum is q_t^{reg}.

A.5 Reward Normalization and Clipping Ablation

We compare three reward-processing configurations to examine the effect of normalization and clipping. All configurations use the same reward combination, initialization, prompts, sampler, reference coefficient, and training budget. "Global std" and "group std" denote global and within-group standard-deviation normalization, respectively; clip is the clipping bound.

With global standard deviation, increasing the clipping bound from 0.5 to 1.0 reduces the overall results. At clip $= 1.0$, group-standard-deviation normalization gives higher HPSv2.1, ImageReward, and UnifiedReward. Considering all six metrics, we use group std with clip $= 1.0$ in the main experiment.

A.6 Additional Experimental Details

Trainable adapters use the low-rank parameterization introduced by LoRA (Hu et al., 2022). The main evaluation uses the Pick-a-Pic prompts (Kirstain et al., 2023) and reports CLIPScore and HPSv2.1 following their original definitions (Hessel et al., 2021; Wu et al., 2023). RISE-DMD uses equal PickScore, CLIPScore, and HPSv2.1 rewards, with 24 samples per prompt group and 48 prompt groups per round. The main setting uses group-standard-deviation normalization, clip $= 1$, and clipped preference scaling.

A.7 Gaussian Forward-Process Identity

Sections 4.3 and 4.4 use the velocity–score relation under a shared Gaussian forward process. Let

$$x_t = \alpha_t x_0 + \sigma_t \epsilon, \qquad \epsilon \sim \mathcal{N}(0, I), \qquad u_t = \dot{\alpha}_t x_0 + \dot{\sigma}_t \epsilon.$$

For any distribution generated by this forward process, let $v_t(x_t) = \mathbb{E}[u_t \mid x_t]$ denote its marginal velocity field. Since $\epsilon = (x_t - \alpha_t x_0)/\sigma_t$, we have

$$v_t = \frac{\dot{\sigma}_t}{\sigma_t} x_t + \left(\dot{\alpha}_t - \frac{\alpha_t \dot{\sigma}_t}{\sigma_t} \right) \mathbb{E}[x_0 \mid x_t].$$

The Gaussian posterior score is

$$\nabla_{x_t} \log p_t(x_t) = \frac{\alpha_t \mathbb{E}[x_0 \mid x_t] - x_t}{\sigma_t^2}.$$

Eliminating $\mathbb{E}[x_0 \mid x_t]$ gives the velocity–score relation for one marginal:

$$v_t^p(x_t) = \frac{\dot{\alpha}_t}{\alpha_t} x_t + \xi_t \nabla_{x_t} \log p_t(x_t).$$

Therefore, for two distributions A and B following the same forward process, the shared drift term cancels and

$$v_t^A - v_t^B = \xi_t \nabla_{x_t} \log \frac{p_t^A(x_t)}{p_t^B(x_t)},$$

where

$$\xi_t = \frac{\sigma_t^2}{\alpha_t} \left(\dot{\alpha}_t - \frac{\alpha_t \dot{\sigma}_t}{\sigma_t} \right)$$

depends only on the noise schedule of the forward process. This coefficient only relates the score and marginal velocity field under the shared Gaussian forward process; it is not an additional model parameter.

Let the current on-policy marginal velocity field be $v_{\bar{\theta}}$, and define $\bar{r}_t(x_t) = \mathbb{E}[r(x_0) \mid x_t]$. Using the conditional density induced by the Gaussian kernel, differentiation under the integral gives

$$\mathbb{E}[(u_t - v_{\bar{\theta}}) r(x_0) \mid x_t] = \xi_t \nabla_{x_t} \bar{r}_t(x_t).$$

Because $\mathbb{E}[u_t - v_{\bar{\theta}} \mid x_t] = 0$, this implies

$$\mathbb{E}[(2r - 1)(u_t - v_{\bar{\theta}}) \mid x_t] = 2\xi_t \nabla_{x_t} \bar{r}_t(x_t),$$

which is the identity used in Section 4.3.

A.8 Old-Field Mismatch

In practice, Reward Score uses an EMA field as $v_{\bar{\theta}}$, which need not exactly equal the current rollout field. Let

$$h = v_{\bar{\theta}}, \qquad e = h - v_\phi, \qquad m = \mathbb{E}_{p_\phi}[r \mid x_t].$$

Using the same squared objective and centering the positive and negative branches at h, let $v^*_{\mathrm{reg}}(h)$ denote the population-optimal output of the full Reward Score objective when the previous field is h. Direct expansion gives

$$v^*_{\mathrm{reg}}(h) = v^*_{\mathrm{reg}}(v_\phi) + \frac{\beta - (2m - 1)}{\beta + \lambda_{\mathrm{ref}}} e.$$

Since $r \in [0, 1]$,

$$\left\| v^*_{\mathrm{reg}}(h) - v^*_{\mathrm{reg}}(v_\phi) \right\| \le \frac{\beta + 1}{\beta + \lambda_{\mathrm{ref}}} \|h - v_\phi\|.$$

This result shows that previous-field mismatch directly affects the reward-field target. Under ideal, sufficiently accurate DMD fitting, the student matches the previous Reward Score EMA, so $v_{\bar{\theta}} = v_\phi$ and the EMA can directly serve as the previous field in the next round. Using the fake score instead requires another estimate of the current student field, whose online fitting error enters the new reward-field target and may propagate and accumulate across rounds. With finite training, projection and tracking errors remain between the EMA and the student field; Section 5.3 compares the two choices in a controlled experiment.

A.9 An Idealized Multi-Round Special Case

Under an additional cross-time compatibility assumption, one can derive a clean-space multi-round recursion. The assumption requires that the time-indexed target marginals in each round are all

generated by one clean distribution and that DMD exactly matches that distribution. With a fixed reward transform, the clean endpoint update is

$$p_{\phi_{k+1}}(x_0) \propto p_{\phi_k}(x_0)^{\rho} p_{\mathrm{ref}}(x_0)^{1-\rho} \exp\left(\frac{2}{\beta+\lambda_{\mathrm{ref}}} r(x_0)\right).$$

After K rounds,

$$p_{\phi_K} \propto p_{\phi_0}^{\rho^K} p_{\mathrm{ref}}^{1-\rho^K} \exp\left(\frac{2}{\beta+\lambda_{\mathrm{ref}}} \frac{1-\rho^K}{1-\rho} r\right).$$

For $\lambda_{\mathrm{ref}} > 0$, $0 < \rho < 1$, and under these idealized assumptions,

$$p_{\phi_K} \longrightarrow p_{\mathrm{ref}} \exp\left(\frac{2}{\lambda_{\mathrm{ref}}} r\right) / Z.$$

This result describes only an idealized clean-space iteration under the additional compatibility and exact-projection assumptions. The main theoretical results for RISE-DMD are the population-optimal field, the time-indexed target marginal, and the regularized reward objective in Sections 4.3–4.5; this appendix result does not provide a global convergence guarantee for the practical training algorithm.

B Additional Matched-Input Comparisons

We provide additional Pick-a-Pic qualitative results beyond Figure 3. Figures 6 and 7 show matched-input examples. Within each column, DMD, RT-DM, and RISE-DMD use the same prompt, random seed, and initial noise, and all images are generated with four NFE.

B.1 Weighted Flow Matching Failure Cases

We provide matched-input qualitative results for the weighted Flow Matching ablation in Section 5.5. Each column uses the same prompt and initial noise for weighted Flow Matching and RISE-DMD; the comparison is shown in Figure 8.

C Additional OCR Reward Curves

Figure 9 shows the online OCR reward under two Reward Score EMA schedules. Type-1 improves faster but later degrades; Type-2 improves more slowly and remains stable over the longer training interval.



Figure 6: Matched-input comparison of DMD, RT-DM, and RISE-DMD at four NFE.



Figure 7: Matched-input comparison of DMD, RT-DM, and RISE-DMD at four NFE.

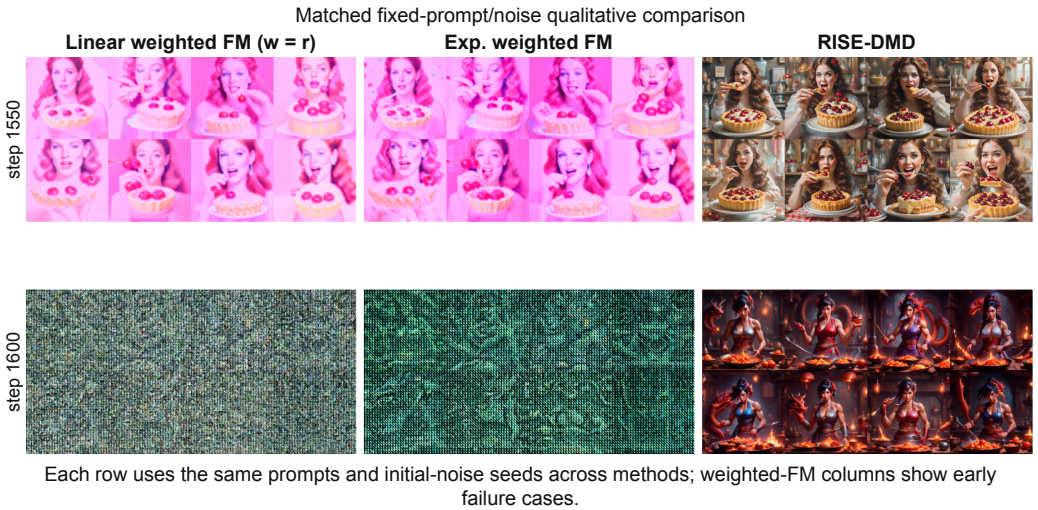

Figure 8: Matched-input comparison of weighted Flow Matching and RISE-DMD.

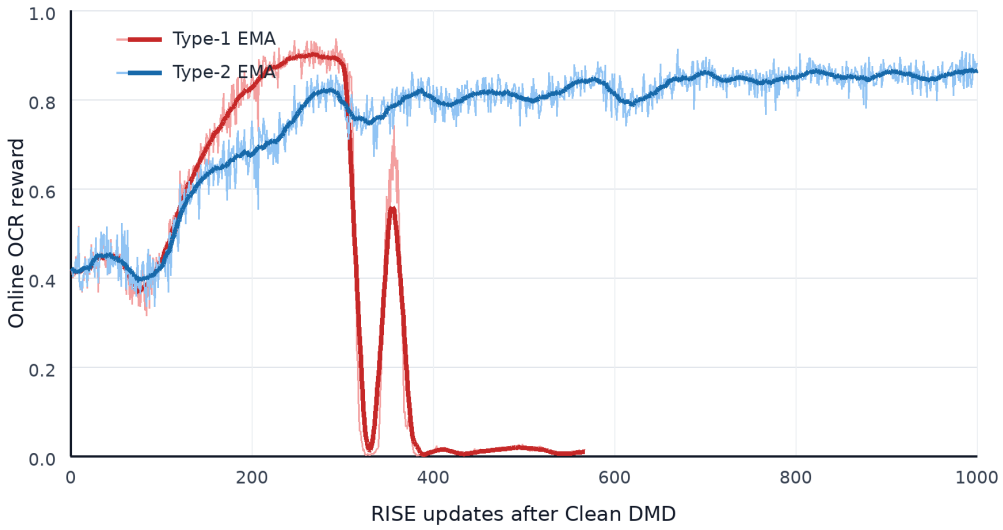

Figure 9: Online OCR reward under two Reward Score EMA schedules.