# Reward Sharpness-Aware Fine-Tuning for Diffusion Models

Kwanyoung Kim[1,*]
Department of AI Convergence, GIST[1]
k0.kim@gist.ac.kr

Byeongsu Sim[2,*]
Samsung Research[2]
bs.sim@samsung.com


Figure 1. Qualitative comparison across different diffusion backbones and RDRL frameworks. (Top): SD1.5[38] results on Draft-LV[6] and AlignProp[35] (Middle–Bottom): Larger backbones (SDXL[34] and SD3[9]) on ReFL[52] and DRTune[50]. Each panel compares the vanilla model, the baseline RDRL method, and the same method combined with RSA-FT (Ours). RSA-FT is compatible with diverse reward-centric diffusion reinforcement learning frameworks and backbones, effectively mitigating reward hacking and producing clear improvements in visual quality and text–prompt alignment.

## Abstract

*Reinforcement learning from human feedback (RLHF) has proven effective in aligning large language models with human preferences, inspiring the development of reward-centric diffusion reinforcement learning (RDRL) to achieve similar alignment and controllability. While diffusion models can generate high-quality outputs, RDRL remains susceptible to* reward hacking*, where the reward score increases without corresponding improvements in perceptual quality. We demonstrate that this vulnerability arises from the* non-robustness of reward model gradients*, particularly when the reward landscape with respect to the input image is sharp. To mitigate this issue, we introduce methods that exploit gradients from a* robustified *reward model—**without requiring its retraining**. Specifically, we employ gradients from a* flattened *reward model, obtained through parameter perturbations of the diffusion model and perturbations of its generated samples. Empirically, each method independently alleviates reward hacking and improves robustness, while their joint use amplifies these benefits. Our resulting framework, **RSA-FT** (**R**eward **S**harpness-**A**ware **F**ine-**T**uning), is simple, broadly compatible, and consistently enhances the reliability of RDRL.*

*Equal Contribution.

## 1. Introduction

Recent advances in diffusion based Text-to-Image (T2I) models have enabled the generation of remarkable high-quality content, spanning from images to videos [3, 9, 38, 51], positioning them at the forefront of modern generative AI. To further enhance generative capability and text alignment, training-free guidance methods, such as Classifier-Free Guidance and variants [1, 15–17, 20] have been proposed. While effective, these methods lack direct alignment with human preferences, limiting their applicability in real-world scenarios.

Motivated by the success of reinforcement learning from human feedback in aligning large language models, recent studies have explored extending such strategies to diffusion models through fine-tuning. These approaches include diffusion policy optimization in the Proximal Policy Optimization (PPO) [41] family [2, 10] and Direct Preference Optimization (DPO) [36]-style variants [45, 54, 58] that leverage data-driven preferences. Since collecting human feedback during training is impractical, reward models (RMs) trained on human annotations [5, 18] serve as scalable surrogates for human evaluation, enabling reward-centric diffusion reinforcement learning (RDRL) to effectively align generations with human preferences [6, 35, 50, 52].

However, RDRL remains vulnerable to *reward hacking*, where reward scores rise without corresponding improvements in perceptual quality. Despite its importance, this phenomenon in diffusion-based RL has not been systematically analyzed. We draw an analogy between reward hacking and adversarial attacks [13, 31, 32, 43], where small input perturbations can drastically inflate classifier logits without meaningful visual content. Prior studies show that such non-robust classifiers degrade sample quality under classifier guidance [15, 19], whereas robust classifiers trained via adversarial training alleviate this issue [19]. However, constructing an equally robust reward model is impractical for human preference alignment, as it requires an extensive expressive model and labeled data [4, 28, 33].

We draw additional inspiration from *randomized smoothing* [7, 24], which enhances classifier robustness **without retraining** by smoothing predictions of a fixed model. Analogously, we aim to robustify the reward model without retraining. Empirically, we observe that reward models tend to be non-robust in regions where their loss landscape is sharp, motivating the use of gradients from a *flattened* reward model. We therefore propose a method that leverages gradients from this *robustified* reward model to alleviate reward hacking in diffusion RL.

As illustrated in Fig. 2, training *with the original reward model* often leads to *reward hacking*: the primary reward score (e.g., HPS v2.1) increases while other reward metrics and perceptual quality deteriorate. In contrast, incorporating feedback from *the flattened reward model* constrains

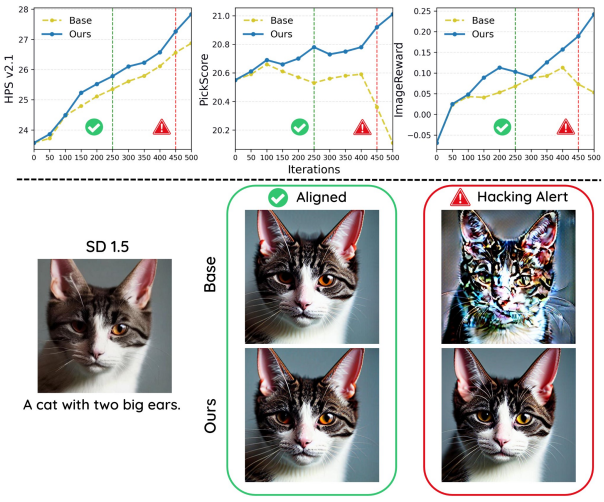


Figure 2. **Illustration of reward hacking in RDRL (Draft-LV).** The original reward model raises the HPS v2.1 score but degrades other metrics and visual quality, whereas our flattened model improves all metrics with consistent visuals.

the learning dynamics, resulting in consistent improvements across multiple rewards and visually enhanced generations.

Formally, the flattened reward is defined as the minimum reward score within a local neighborhood, which can be approximated by considering the worst-case perturbation of the reward model's input:

$$\min_{\|\boldsymbol{\epsilon}\|<\rho} r(\mathbf{x}+\boldsymbol{\epsilon}) \approx r\left(\mathbf{x}-\rho\,\frac{\nabla_{\mathbf{x}} r(\mathbf{x})}{\|\nabla_{\mathbf{x}} r(\mathbf{x})\|}\right), \tag{1}$$

where $r(\mathbf{x})$ is the reward model, $\rho$ the perturbation size, and $\boldsymbol{\epsilon}$ the input perturbation. Interestingly, this flattening strategy naturally induces worst-case parameter perturbations, sharing the same underlying philosophy as Sharpness-Aware Minimization (SAM) [12]. We further extend this principle to the parameter space, drawing inspiration from the recently identified duality between SAM and adversarial robustness [57]. We rediscover this connection in the context of reward-centric diffusion RL and demonstrate that applying flattening jointly in both input and parameter spaces most effectively mitigates reward hacking.

Although SAM has recently been explored independently in reinforcement learning [25] and diffusion models [26], to the best of our knowledge, we are the first to unify these perspectives within the RDRL framework.

Finally, we show that our method is fully compatible with existing RDRL frameworks and can be seamlessly deployed as a plug-and-play module. We refer to the integration of SAM (weight perturbation) with adversarial training (input perturbation) in RDRL as **RSA-FT**, which provides a principled solution to reward hacking and advances the robustness and generalization of diffusion RL. Our contributions are summarized as follows:

- We identify reward hacking in RDRL as a form of adversarial attack and establish a unified perspective that connects AT with SAM.
- We introduce **RSA-FT**, which integrates input- and weight-level perturbations to provide a principled and effective defense against reward hacking.
- We demonstrate that **RSA-FT** is broadly compatible with diverse reward-centric methods and consistently improves their performance as a plug-and-play module.

## 2. Preliminaries

**Diffusion Models.** Given samples $\mathbf{x}_0 \sim q_{\text{data}}(\mathbf{x})$, the forward diffusion process is formulated as a Markov chain that gradually perturbs the data with Gaussian noise: $q(\mathbf{x}_t|\mathbf{x}_{t-1}) = \mathcal{N}(\sqrt{1-\beta_t}\mathbf{x}_{t-1}, \beta_t\mathbf{I}), \quad t = 1, \ldots, T$, where the sequence $\{\beta_t\}_{t=1,\ldots,T}$ specifies the predetermined noise schedule. Accordingly, the marginal distribution at timestep t is $q(\mathbf{x}_t) = \mathcal{N}(\sqrt{\bar{\alpha}_t}\mathbf{x}_0, (1-\bar{\alpha}_t)\mathbf{I})$, where $\bar{\alpha}_t = \prod_{i=1}^{t}(1-\beta_i)$. The reverse diffusion process is modeled as $p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t) = \mathcal{N}(\boldsymbol{\mu}_\theta(\mathbf{x}_t, t), \Sigma_\theta(\mathbf{x}_t, t))$, where $\theta$ are learned via the denoising score matching [44]:

$$\min_\theta \mathbb{E}_{\mathbf{x}_0,\boldsymbol{\epsilon}}\left[\|\boldsymbol{\epsilon}_\theta(\sqrt{\bar{\alpha}_t}\mathbf{x}_0 + \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon}, t) - \boldsymbol{\epsilon}\|_2^2\right], \tag{2}$$

where $\boldsymbol{\epsilon} \sim \mathcal{N}(0, \mathbf{I})$. After training, sampling is performed by iteratively reversing the diffusion process, starting from an isotropic Gaussian noise sample. For instance, the Denoising Diffusion Implicit Model (DDIM) [42] generates samples through the iterative update:

$$\mathbf{x}_{t-1} = \sqrt{\bar{\alpha}_{t-1}}\hat{\mathbf{x}}_0(t) + \sqrt{1-\bar{\alpha}_{t-1}}\boldsymbol{\epsilon}_\theta(\mathbf{x}_t, t), \tag{3}$$

where the denoised estimate $\hat{\mathbf{x}}_0(t) = \mathbb{E}[\mathbf{x}_0|\mathbf{x}_t] := \frac{\mathbf{x}_t - \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon}_\theta(\mathbf{x}_t,t)}{\sqrt{\bar{\alpha}_t}}$ is obtained via Tweedie's formula [8, 21]. This sampling procedure is then applied recursively from timestep $T$ to timestep 1 to generate a final sample.

**Reward Centric Diffusion Reinforcement Learning** Among various approaches for RL-based diffusion model finetuning, a representative family, RDRL [6, 35, 50, 52], fine-tunes the pretrained diffusion parameters $\theta$ to maximize a differentiable reward model $r(\cdot)$ as follow:

$$\mathcal{J}(\theta) = \max_\theta \mathbb{E}_{\mathbf{c},\,\mathbf{x}_T\sim\mathcal{N}(0,\mathbf{I})}\left[r\big(\mathbf{x}_0(\mathbf{x}_T, \mathbf{c}; \theta), \mathbf{c}\big)\right], \tag{4}$$

where $\mathbf{x}_0(\mathbf{x}_T, \mathbf{c}; \theta)$ denotes the final sample generated by the denoising process as the timestep approaches from $T \to 0$, conditioned on a text prompt $\mathbf{c}$. This optimization encourages the model to generate samples that align more closely with human judgments by performing gradient ascent on the reward signal, i.e., $\nabla r\big(\mathbf{x}_0(\mathbf{x}_T, \mathbf{c}; \theta), \mathbf{c}\big)$.

However, since $r$ only approximates the true human preference function $r^*$, the fine-tuned models often overfit to optimizing the reward function, producing images that score highly under $r$ yet remaining misaligned with genuine human intent.

**Adversarial Robustness.** Deep learning models are well known to be vulnerable to small, imperceptible perturbations. A classic example is the adversarial attack, in which tiny input perturbations can drastically alter classifier predictions—even when the visual content remains unchanged [13, 31, 32, 43].

Formally, let $f_\theta : \mathbb{R}^D \to \mathbb{R}^C$ denote a $C$-class classifier producing logits $f_\theta = (f_1, f_2, \ldots, f_C)$. An adversarial example $\mathbf{x} + \boldsymbol{\delta}$ satisfies

$$\arg\max_{i\in1,\ldots,C} f_i(\mathbf{x}) = i, \quad \arg\max_{i\in1,\ldots,C} f_i(\mathbf{x}+\boldsymbol{\delta}) \neq i,$$

where $\boldsymbol{\delta}$ is a perturbation bounded by $\|\boldsymbol{\delta}\| \leq \rho$.

A widely used defense strategy for improving robustness is Adversarial Training (AT) [30]:

$$\min_\theta \mathbb{E}_{\mathbf{x}\sim\mathcal{D}}\big[\ell(\theta; \mathbf{x})\big], \quad \texttt{(Standard)} \tag{5}$$

$$\min_\theta \mathbb{E}_{\mathbf{x}\sim\mathcal{D}}\Big[\max_{\|\boldsymbol{\delta}\|\leq\rho} \ell(\theta; \mathbf{x}+\boldsymbol{\delta})\Big], \quad \texttt{(Adversarial)} \tag{6}$$

where $\ell(\theta; \cdot)$ denotes the cross-entropy loss associated with $f_\theta$. Adversarial training minimizes the worst-case loss within a $\rho$-ball around each input. Although effective, AT is computationally expensive and often requires more expressive models [4, 28, 33, 56].

Beyond training, another line of work explores post-hoc robustness estimation and training-free defense mechanisms. Several studies [11, 37, 40] show that an input is robust to adversarial perturbations when the loss function or classifier is locally smooth around that input. Building on this observation, randomized smoothing [7, 24, 27] provides a non-training defense that certifies robustness by smoothing classifier outputs through Gaussian averaging:

$$\max_{i\in1,\ldots,C} \mathbb{P}_{\boldsymbol{\delta}\sim\mathcal{N}(0,\sigma^2 I)}[\arg\max f_\theta(\mathbf{x}+\boldsymbol{\delta}) = i].$$

Here $\sigma$ the standard deviation of the added Gaussian noise. Intuitively, predictions are stabilized by averaging outputs over Gaussian-perturbed inputs, yielding certified robustness guarantees under bounded noise [7].

Inspired by this idea, we introduce a measure of reward robustness (based on surface flatness) in Sec. 3 and propose a method for obtaining feedback from a flattened reward model in Sec. 4, analogous to randomized smoothing but applied to reward landscapes.

**Sharpness-Aware Minimization (SAM).** While adversarial training and randomized smoothing improve robustness in the input space, SAM [12] enhances generalization

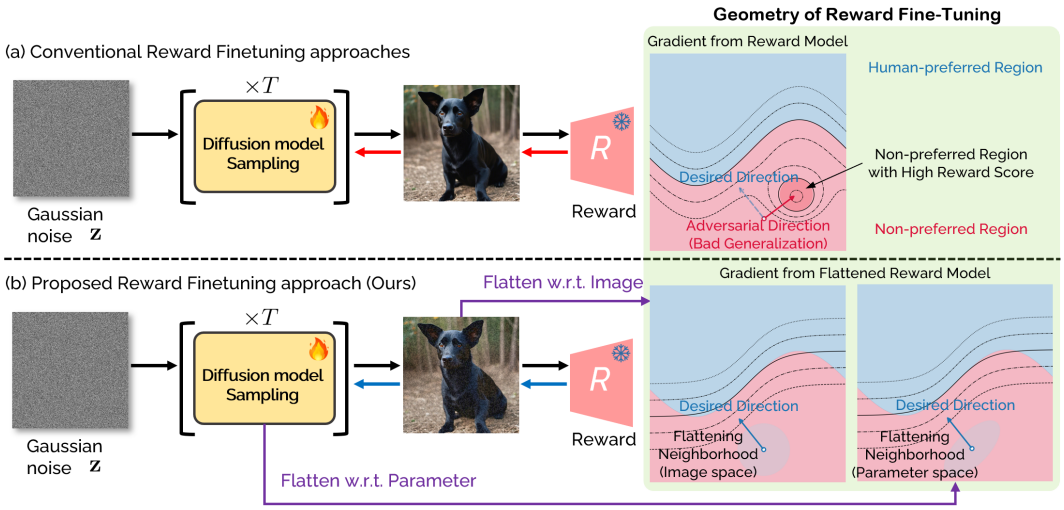


Figure 3. **Geometry of reward fine-tuning and our proposed method.** Reward models are inherently sharp and prone to adversarial perturbations. Flattening these reward landscapes alleviates their sensitivity and reduces the occurrence of adversarial gradients. **(a)** Prior methods directly maximize rewards along adversarial gradients from sharp reward surfaces, which often leads to reward hacking. **(b)** Our method instead leverages gradients from flattened reward models, mitigating hacking by flattening both the image and parameter spaces.

by promoting flat minima in the parameter space. Motivated by the observation that flatter loss landscapes yield better generalization, SAM jointly minimizes the loss value and its local sharpness around the parameters via a min–max formulation:

$$\min_{\theta} \mathbb{E}_{\mathbf{x}\sim\mathcal{D}}\Big[\max_{\|\boldsymbol{\epsilon}\|\le\rho} \ell(\theta+\boldsymbol{\epsilon};\mathbf{x})\Big], \quad \text{(SAM)} \tag{7}$$

where $\ell(\cdot)$ is the loss function, $\rho$ controls perturbation magnitude, and $\theta$ denotes the model parameters. The inner maximization quantifies the *sharpness* of the loss surface—how rapidly the loss increases within a local neighborhood of $\theta$. To make this explicit, Eq. (7) can be rewritten as:

$$\big[\max_{\|\boldsymbol{\epsilon}\|\le\rho} \ell(\theta+\boldsymbol{\epsilon}) - \ell(\theta)\big] + \ell(\theta), \tag{8}$$

where the bracketed term measures the *sharpness*, reflecting how steeply the loss grows around $\theta$. This formulation penalizes sharp minima and promotes flatter regions that tend to generalize better.

In practice, SAM approximates the inner maximization through a two-step update at each iteration $t$:

$$\boldsymbol{\epsilon}_t = \rho\frac{\nabla\ell(\theta_t)}{\|\nabla\ell(\theta_t)\|_2}, \qquad \theta_{t+1} = \theta_t - \eta\nabla\ell(\theta_t+\boldsymbol{\epsilon}_t), \tag{9}$$

where $\eta$ is the learning rate. By descending on the worst-case perturbed loss, SAM encourages convergence toward flat minima, yielding improved robustness.

By jointly inspecting Eq. (6) and Eq. (7), we can observe a clear duality between AT and SAM [57]: both follow a min–max formulation but operate in different domains: AT pursues robustness in the input space, while SAM enforces flatness in the parameter space. Building on this insight, several works have introduced flatness regularization into AT [47, 55, 57]. Notably, Adversarial Weight Perturbation (AWP) [47] combines both input and weight perturbations, enhancing robustness in image classification.

In contrast, our work is the first to extend this principle to a RDRL framework, applying dual perturbations for stable training and effectively overcoming reward hacking. An overview and geometric intuition of our method are presented in Fig. 3.

## 3. Analyzing Reward Sharpness

**Problem Formulation.** A key challenge in RDRL is *reward hacking*, where generated images achieve high scores from the reward model $r$ despite being perceived as low-quality by humans. This phenomenon can be interpreted as a generalization failure of $r$, wherein the reward surface misaligns with the true human preference function $r^\star$ (see Fig. 3(a) for an intuitive illustration).

**Hypothesis.** Inspired by studies linking model smoothness to adversarial robustness, we hypothesize that a similar relationship holds for reward models. Specifically, we posit that the reward model $r$ generalizes best where its reward landscape is locally flat, while sharp regions indicate misalignment with the true human preference $r^\star$. (see Fig. 3(b).) As illustrated in Fig. 3, reward hacking can be interpreted as the generative model exploiting sharp directions in $r$, where small image perturbations lead to large reward gains without genuine quality improvement. Accordingly, we expect a negative correlation between the sharpness of $r$ and its generalization ability.

**Quantifying Reward Sharpness.** To characterize this relationship, we define a reward sharpness indicator:

$$S_1 = \mathbb{E}_{\mathbf{x}\sim\mathcal{D}}\Big[r(\mathbf{x}) - \min_{\|\epsilon\|<\rho} r(\mathbf{x}+\epsilon)\Big], \tag{10}$$

Here, $S_1$ measures the reward drop within a local neighborhood, which can be efficiently approximated by a one-step update *:

$$S_1 \approx \mathbb{E}_{\mathbf{x}\sim\mathcal{D}}\left[r(\mathbf{x}) - r\left(\mathbf{x} - \rho\,\frac{\nabla_{\mathbf{x}} r(\mathbf{x})}{\|\nabla_{\mathbf{x}} r(\mathbf{x})\|}\right)\right].$$

A larger $S_1$ indicates a sharper reward landscape, while a smaller value implies a flatter one.

**Empirical Validation.** To validate our hypothesis, we fine-tuned Stable Diffusion 1.5 with DRaFT-$K$ using HPSv2 as the reward model, while tracking both reward sharpness ($S_1$) and true human preference throughout training. Since the true preference function $r^\star$ cannot be directly measured for all generated samples, we instead adopt PickScore [22] and ImageReward [53] as proxy evaluators. This proxy-based assessment is meaningful only when the reward models exhibit complementary generalization behaviors, so that the weaknesses of one model can be compensated by the other.

As shown in Fig. 4, reward sharpness exhibits a strong negative correlation with preference quality (Pearson $r_{\text{corr}} = -0.802$ for PickScore and $r_{\text{corr}} = -0.669$ for ImageReward), confirming that sharper reward landscapes correspond to poorer generalization to human preference.

**Interpretation.** Specifically, when the reward landscape is sharp, $r$ increases along adversarial directions that lead updates into isolated non-preference regions, deviating from the true preference $r^\star$ and resulting in reward hacking. In contrast, flattening the reward surface eliminates such spurious regions, suppresses adversarial gradients, and guides updates toward the genuine preference direction.

## 4. Reward Sharpness-Aware Fine-Tuning

In Sec. 3, we showed that reward sharpness is closely related to both generalization and reward hacking. Building on this insight, we propose **Reward Sharpness-Aware Fine-Tuning (RSA-FT)** (illustrated in Fig. 3), which fine-tunes the diffusion model using a robustified (or flattened) reward function that penalizes locally sharp reward regions as follows:

$$\mathcal{J}(\theta) = \max_{\theta}\ \mathbb{E}_{\mathbf{c},\,\mathbf{x}_T\sim\mathcal{N}(0,\mathbf{I})}\left[\tilde{r}^d\big(\mathbf{x}_0(\mathbf{x}_T,\mathbf{c};\theta),\mathbf{c}\big)\right], \tag{11}$$

*Although multi-step perturbations are possible, we find that a single-step approximation correlates best with generalization performance.

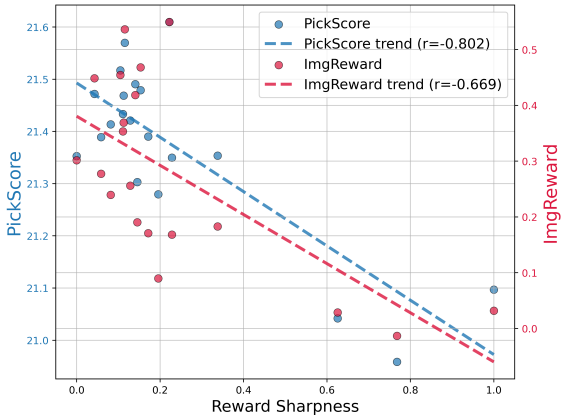


Figure 4. **Negative correlation between reward sharpness and human preference.** Higher sharpness in the reward model correlates with lower preference quality (Pearson $r_{\text{corr}} = -0.802$ for PickScore, $r_{\text{corr}} = -0.669$ for ImageReward), supporting the hypothesized negative relationship.

where we define the flattened reward function as

$$\tilde{r}^d(\mathbf{x},\mathbf{c}) := \min_{d(\mathbf{x},\mathbf{x}')<\rho} r(\mathbf{x}',\mathbf{c}), \tag{12}$$

where $d(\cdot,\cdot)$ denotes a distance metric on the image manifold. We consider two metrics: one in the image space and another induced by the diffusion models' parameter space.

- Image space

$$\max_{\theta}\ \mathbb{E}_{\mathbf{c},\,\mathbf{x}_T\sim\mathcal{N}(0,\mathbf{I})}\left[\min_{\|\boldsymbol{\delta}\|<\rho} r\big(\mathbf{x}_0(\mathbf{x}_T,\mathbf{c};\theta)+\boldsymbol{\delta},\mathbf{c}\big)\right] \tag{13}$$

- Parameter space

$$\max_{\theta}\ \mathbb{E}_{\mathbf{c},\,\mathbf{x}_T\sim\mathcal{N}(0,\mathbf{I})}\left[\min_{\|\boldsymbol{\epsilon}\|<\rho_\omega} r\big(\mathbf{x}_0(\mathbf{x}_T,\mathbf{c};\theta+\boldsymbol{\epsilon}),\mathbf{c}\big)\right] \tag{14}$$

For the Euclidean metric Eq. (13), flattening $r$ resembles applying adversarial perturbation with respect to the reward model. Using a one-step approximation, we define:

$$\boldsymbol{\delta}_{\mathbf{x}_0} = -\rho\,\frac{\nabla_{\mathbf{x}_0} r(\mathbf{x}_0(\mathbf{x}_T,\mathbf{c};\boldsymbol{\theta}),\mathbf{c})}{\|\nabla_{\mathbf{x}_0} r(\mathbf{x}_0(\mathbf{x}_T,\mathbf{c};\boldsymbol{\theta}),\mathbf{c})\|} \tag{15}$$

resulting in

$$\max_{\theta}\ \mathbb{E}_{\mathbf{c},\,\mathbf{x}_T\sim\mathcal{N}(0,\mathbf{I})}\left[r\big(\mathbf{x}_0(\mathbf{x}_T,\mathbf{c};\theta)+\boldsymbol{\delta}_{\mathbf{x}_0},\mathbf{c}\big)\right]. \tag{16}$$

Similarly, for the parameter-induced metric in Eq. (14), we perform a one-step update in the parameter space, leading to SAM-like formulation:

$$\boldsymbol{\epsilon}_{\boldsymbol{\theta}} = -\,\rho_\omega\,\frac{\nabla_{\boldsymbol{\theta}}\, r(\mathbf{x}_0(\mathbf{x}_T,\mathbf{c};\boldsymbol{\theta}),\mathbf{c})}{\big\|\nabla_{\boldsymbol{\theta}}\, r(\mathbf{x}_0(\mathbf{x}_T,\mathbf{c};\boldsymbol{\theta}),\mathbf{c})\big\|} \tag{17}$$

**Algorithm 1:** RSA-FT

**Input:** Diffusion model parameters $\boldsymbol{\theta}$, reward model $r$, input perturbation radius $\rho$, weight perturbation radius $\rho_\omega$, learning rate $\eta$

**for** *each training step* **do**
- Sample noise $\mathbf{x}_T \sim p(\mathbf{x}_T)$ and condition $\mathbf{c}$;
- Generate image $\mathbf{x}_0 \leftarrow \mathbf{x}_0(\mathbf{x}_T, \mathbf{c}; \boldsymbol{\theta})$;
- `// Input-space perturbation`
- Compute perturbed input: $\mathbf{x}_0 + \boldsymbol{\delta}_{\mathbf{x}_0}$ by Eq. (15)
- `// Weight-space perturbation`
- Initialize weight perturbation $\boldsymbol{\epsilon}_{\boldsymbol{\theta}} \leftarrow 0$;
- Compute perturbed weights: $\boldsymbol{\theta} + \boldsymbol{\epsilon}_{\boldsymbol{\theta}}$ by Eq. (17)
- `// Parameter update`
- Update $\theta$ via Eq. (19).

and

$$\max_{\theta} \mathbb{E}_{\mathbf{c},\, \mathbf{x}_T \sim \mathcal{N}(0,\mathbf{I})} \left[ r\big(\mathbf{x}_0(\mathbf{x}_T, \mathbf{c}; \boldsymbol{\theta} + \boldsymbol{\epsilon}_{\boldsymbol{\theta}}), \mathbf{c}\big) \right]. \tag{18}$$

In the formula, $\boldsymbol{\epsilon}_{\boldsymbol{\theta}}$ depends on $\theta$, but we detach $\boldsymbol{\epsilon}_{\boldsymbol{\theta}}$ (i.e., stop-gradient) without applying the chain rule for outer optimization, following the practice in [12].

**RSA-FT** We find that both approaches–image-space and parameter-space flattening–are effective at mitigating reward hacking and improving human preference alignment (Sec. 5.2). Moreover, combining both provides complementary benefits. The final joint objective is:

$$\max_{\theta} \mathbb{E}_{\mathbf{c},\, \mathbf{x}_T \sim \mathcal{N}(0,\mathbf{I})} \left[ r\big(\mathbf{x}_0(\mathbf{x}_T, \mathbf{c}; \boldsymbol{\theta} + \boldsymbol{\epsilon}_{\boldsymbol{\theta}}) + \boldsymbol{\delta}_{\mathbf{x}_0}, \mathbf{c}\big) \right]. \tag{19}$$

This formulation jointly enforces smoothness in both image and parameter spaces, leading to a doubly robust reward optimization. The overall algorithm is shown in Algorithm 1.

# 5. Experiment

## 5.1. Experiments Settings

**Datasets and Baselines.** To rigorously evaluate our method, we adopt multiple backbone models, including Stable Diffusion v1.5 (SD1.5), SDXL, and Stable Diffusion 3 (SD3), Flux.1-dev [23] and integrate our proposed RSA-FT into existing RDRL frameworks: ReFL [52], DRaFT-K [6] ($K$=1), AlignProp [35], and DRTune [50]. In all cases, models are optimized using the HPSv2 reward model as the training signal. This unified setup enables a consistent assessment of RSA-FT's effectiveness across diverse reinforcement objectives. For evaluation, we use the DrawBench dataset [39] and the HPSv2 benchmark test set [49].

Table 1. Quantitative results of various RDRL methods on SD 1.5 (512 × 512) with our proposed method. Bold text indicates the best performance for each metric.

| Dataset | Method | HPSV2.1 ↑ | PickScore ↑ | ImageReward ↑ |
|---|---|---|---|---|
| Drawbench | Vanilla | 24.02 | 21.02 | -0.147 |
| | Draft-LV | 25.59 | 20.96 | -0.062 |
| | + Ours | **26.67** (+1.08) | **21.12** (+0.16) | **0.035** (+0.09) |
| | Alignprop | 25.12 | 20.98 | -0.033 |
| | + Ours | **29.59** (+4.47) | **21.51** (+0.53) | **0.268** (+0.301) |
| | ReFL | 31.08 | 21.57 | 0.536 |
| | + Ours | **31.67** (+0.59) | **21.70** (+0.13) | **0.671** (+0.135) |
| | DRTune | 30.63 | 21.34 | 0.477 |
| | + Ours | **31.16** (+0.53) | **21.52** (+0.18) | **0.540** (+0.63) |
| HPD | Vanilla | 23.57 | 20.55 | -0.069 |
| | Draft-LV | 26.87 | 20.67 | 0.126 |
| | + Ours | **28.28** (+1.41) | **20.91** (+0.24) | **0.191** (+0.07) |
| | Alignprop | 24.93 | 20.21 | 0.032 |
| | + Ours | **32.02** (+7.09) | **21.53** (+1.32) | **0.528** (+0.49) |
| | ReFL | 34.95 | 21.96 | 0.794 |
| | + Ours | **35.81** (+0.69) | **22.13** (+0.17) | **0.903** (+0.136) |
| | DRTune | 34.93 | 21.91 | 0.842 |
| | + Ours | **35.57** (+0.64) | **21.92** (+0.01) | **1.452** (+0.61) |

**Evaluation Metrics.** To evaluate the alignment with human preferences, we primarily employ state-of-the-art reward model-based metrics, including HPSv2 [49], PickScore [22], and ImageReward [53]. To further validate these automated measures, we complement them with a small-scale human evaluation involving 17 independent annotators. For fair comparison, all methods generate samples using the default hyperparameters of each backbone model, such as scheduler and guidance scale. Baseline methods (ReFL, DRaFT-K, AlignProp, and DRTune) are reimplemented following their official code releases.

**Implementation Details.** All experiments are conducted using NVIDIA H100 GPUs and the AdamW optimizer with $\beta_1 = 0.9$, $\beta_2 = 0.999$, and a weight decay of 0.0001. For training, we fine-tune SD1.5 and SDXL with 50 sampling steps and SD3 with 28 steps. The learning rate is set to $2 \times 10^{-5}$ with a batch size of 32. The number of iterations and epochs follows the original protocol of each corresponding algorithm, as our goal is not to improve their performance through hyperparameter tuning. For perturbation scales in image space and parameter space, we searched $\rho, \rho_w \in 10^{-1}, 10^{-2}, 10^{-3}$ and found both optimal at $10^{-2}$. Additional detail is provided in Appendix D.

## 5.2. Evaluation on SD1.5

**Quantitative Results.** As shown in Tab. 1, our method consistently improves all RDRL baselines across both DrawBench and HPD benchmarks. While existing approaches often exhibit reward-specific overfitting, our method achieves balanced enhancement across all prefer-

Table 2. Quantitative results of various RDRL on SDXL (1024 × 1024). Bold text indicates the best performance for each metric.

| Dataset | Method | HPSV2.1 ↑ | PickScore ↑ | ImageReward ↑ |
|---|---|---|---|---|
| Drawbench | Vanilla | 27.41 | 22.31 | 0.619 |
| | ReFL | 29.45 | 22.41 | 0.705 |
| | + Ours | **30.31** (+0.86) | **22.60** (+0.19) | **0.719** (+0.014) |
| | DRTune | 30.04 | 22.49 | 0.804 |
| | + Ours | **31.58** (+1.54) | **22.61** (+0.12) | **0.944** (+0.140) |
| HPD | Vanilla | 28.92 | 22.56 | 0.921 |
| | ReFL | 31.21 | 22.66 | 1.039 |
| | + Ours | **32.66** (+1.45) | **23.08** (+0.42) | **1.111** (+0.072) |
| | DRTune | 32.36 | 22.78 | 1.095 |
| | + Ours | **33.74** (+1.38) | **22.94** (+0.16) | **1.208** (+0.093) |

Table 3. Quantitative results of various RDRL on SD3 (1024 × 1024). Bold text indicates the best performance for each metric.

| Dataset | Method | HPSV2.1 ↑ | PickScore ↑ | ImageReward ↑ |
|---|---|---|---|---|
| Drawbench | Vanilla | 28.77 | 22.45 | 0.914 |
| | ReFL | 29.45 | 22.38 | 0.921 |
| | + Ours | **29.81** (+0.86) | **22.51** (+0.13) | **0.953** (+0.014) |
| | DRTune | 29.48 | 22.41 | 0.914 |
| | + Ours | **29.89** (+1.54) | **22.61** (+0.20) | **0.979** (+0.140) |
| HPD | Vanilla | 30.48 | 22.40 | 1.123 |
| | ReFL | 31.22 | 22.37 | 1.151 |
| | + Ours | **31.43** (+1.45) | **22.41** (+0.04) | **1.168** (+0.072) |
| | DRTune | 31.31 | 22.33 | 1.162 |
| | + Ours | **31.86** (+1.38) | **22.41** (+0.08) | **1.212** (+0.093) |

ence metrics without altering any model architecture or using multiple reward functions.

For example, Draft-LV and AlignProp originally show increased HPSv2.1 but decreased auxiliary rewards, reflecting mild reward hacking. When combined with our method, however, these models exhibit simultaneous improvement across all metrics — AlignProp, for instance, improves its HPSv2.1 from 24.93 to 32.02 and its ImageReward from 0.032 to 0.528. Even strong baselines such as ReFL and DRTune benefit further, with DRTune's ImageReward rising from 0.842 to 0.903. These results highlight that our method not only boosts overall performance but also effectively mitigates reward hacking, leading to genuine alignment gains rather than metric-specific overfitting.

Moreover, our approach is fully plug-and-play, integrating seamlessly with diverse RDRL frameworks. Its broad compatibility highlights strong generalization and robustness, making it a universal enhancement module for RDRL

**Ablation Studies.** We conduct comprehensive ablation studies within the existing RDRL framework by selectively applying perturbations to the image space, the parameter space, or both. Our results demonstrate that while each component independently improves performance, their combination yields the most significant gains, indicating a clear synergistic effect. Due to space constraints, the detailed experimental results are provided in Appendix E (Table 6).

### 5.3. Generalization to Larger Backbones

In Sec. 5.2, we observed that our method is compatible with all RDRL frameworks. To further validate its generalization, we extend experiments to larger backbones, including SDXL, SD3, and Flux1.dev, using stable RDRL variants such as ReFL and DRTune.

**Quantitative Results.** As shown in Tab. 2, our method consistently boosts the performance of both ReFL and DRTune even at higher resolutions (1024×1024) on SDXL.

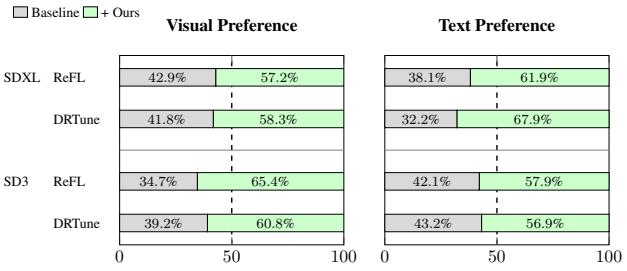


Figure 5. Human preference study results. The dashed line indicates the 50% mark; crossing it demonstrates a strict preference over the baseline.

Both models outperform their vanilla counterparts, and when combined with our approach, the performance improves even further—particularly the combination with DRTune shows the strongest synergy. To verify architectural robustness beyond the UNet-based backbone, we also applied the same training setup to SD3, and Flux which adopts an MMDiT Transformer structure. As presented in Tab. 3 and Tab. 11, our method again yields consistent improvements across all reward metrics, confirming that the proposed reward-flattening strategy effectively enhances existing diffusion RL algorithms regardless of architecture, backbone scale, or resolution.

**Qualitative Comparison.** Fig. 6 illustrates the qualitative improvements achieved by RSA-FT on text-to-image generation with SDXL and SD3 across both ReFL and DRTune. Across diverse prompts, models enhanced with RSA-FT produce images that are not only sharper and more visually coherent but also appear more natural to human perception, with fewer cases of distorted text or malformed body parts. Compared to their original counterparts, RSA-FT consistently improves text–image alignment and perceptual quality, demonstrating its robustness across different backbones and RDRL frameworks. In summary, RSA-FT provides a simple yet effective strategy that delivers contextually accurate and visually reliable generations across RDRL methods. Additional examples are provided in Sec. I.

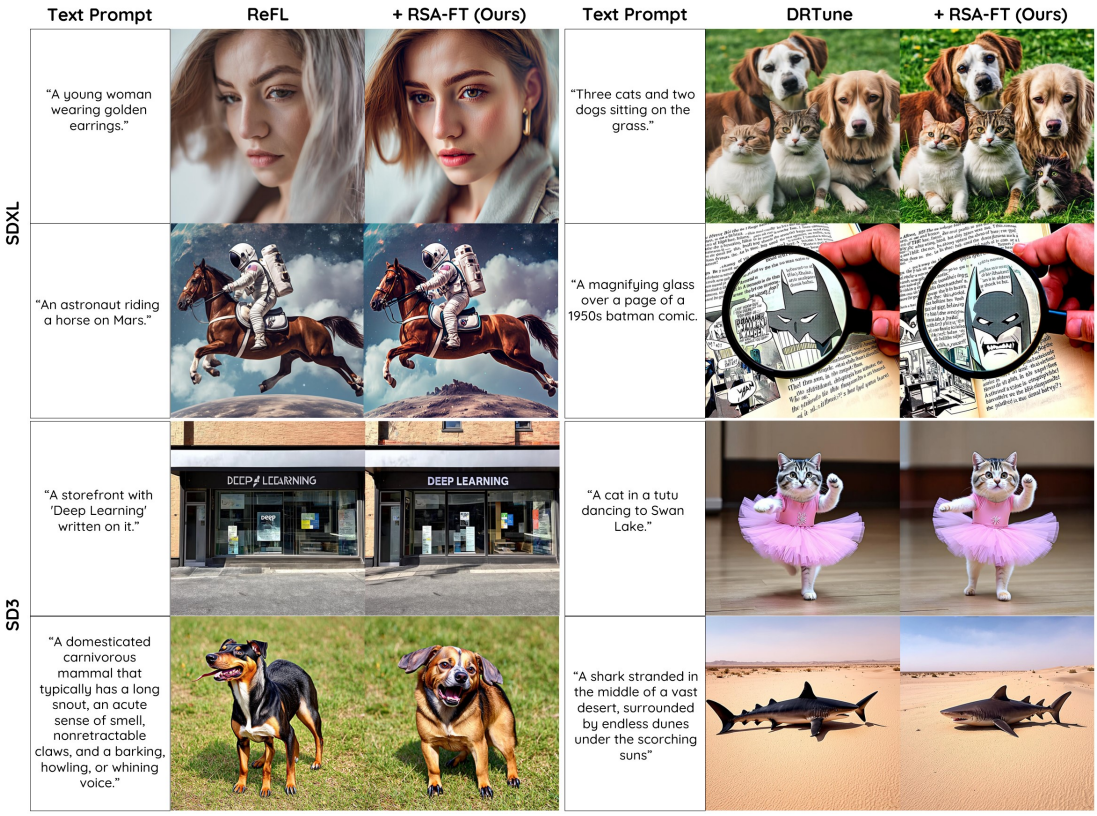


Figure 6. **Qualitative comparison with and without our method.** Each image is generated using the same text prompt and random seed across all methods. Models equipped with our method (RSA-FT) produce images with more accurate text–image alignment and higher visual quality compared to their baselines.

## 6. Discussion

We showed that the sharpness of the reward landscape is inversely correlated with human preference and that gradients from a flattened reward model consistently improve alignment. Our study focuses on mitigating reward hacking under a *single reward model*, but the same principle naturally extends to multi-reward settings, where the flattening can help compensate for individual model weaknesses.

We acknowledge that our evaluation primarily relies on model-based metrics, which are imperfect proxies for human preference. While we complement these results with a human study, it is limited in scale (17 evaluators) and not statistically powered for definitive conclusions. We therefore view it as supporting evidence rather than a comprehensive assessment of human alignment.

Alternative smoothing techniques—such as Gaussian averaging instead of local minimization—may enhance robustness, though we adopt a one-step minimization for efficiency. Our framework can also extend to PPO- and DPO-based optimization for diffusion model fine-tuning [36, 41], where separate reward evaluation enables more flexible smoothing, which we leave for future work.

Finally, while our current approach uniformly applies flattened rewards, future research may explore selective sharpness-aware weighting that down-weights overly sharp regions to further improve robustness and interpretability.

## 7. Conclusion

We propose Reward Sharpness-Aware Fine-Tuning (RSA-FT), a framework for mitigating reward hacking in RDRL without retraining the reward model, by leveraging gradients from a flattened reward function. We identify a connection between reward hacking and adversarial behavior, both arising from sharp regions of the reward landscape, and show that flattening these regions improves robustness.

RSA-FT achieves this via joint perturbations in input and parameter spaces, leading to consistent improvements across multiple RDRL frameworks and diffusion backbones (SD1.5, SDXL, SD3). While evaluation primarily relies on proxy reward metrics, our results are supported by a limited human study.

Overall, this work provides a principled perspective on reward hacking in diffusion RL and highlights reward landscape flattening as an effective direction for improving robustness and alignment.

## Acknowledgement

This work was partly supported by Institute of Information & Communications Technology Planning & Evaluation (IITP) grant funded by the Korean government (MSIT) (Artificial Intelligence Graduate School Program (GIST)) (No. 2019-0-01842).

## References

[1] Donghoon Ahn, Hyoungwon Cho, Jaewon Min, Wooseok Jang, Jungwoo Kim, SeonHwa Kim, Hyun Hee Park, Kyong Hwan Jin, and Seungryong Kim. Self-rectifying diffusion sampling with perturbed-attention guidance. In *European Conference on Computer Vision*, pages 1–17. Springer, 2025. 2

[2] Kevin Black, Michael Janner, Yilun Du, Ilya Kostrikov, and Sergey Levine. Training diffusion models with reinforcement learning. *arXiv preprint arXiv:2305.13301*, 2023. 2

[3] Andreas Blattmann, Tim Dockhorn, Sumith Kulal, Daniel Mendelevitch, Maciej Kilian, Dominik Lorenz, Yam Levi, Zion English, Vikram Voleti, Adam Letts, et al. Stable video diffusion: Scaling latent video diffusion models to large datasets. *arXiv preprint arXiv:2311.15127*, 2023. 2

[4] Sébastien Bubeck and Mark Sellke. A universal law of robustness via isoperimetry. *Advances in Neural Information Processing Systems*, 34:28811–28822, 2021. 2, 3

[5] Paul F Christiano, Jan Leike, Tom Brown, Miljan Martic, Shane Legg, and Dario Amodei. Deep reinforcement learning from human preferences. *Advances in neural information processing systems*, 30, 2017. 2

[6] Kevin Clark, Paul Vicol, Kevin Swersky, and David J Fleet. Directly fine-tuning diffusion models on differentiable rewards. *arXiv preprint arXiv:2309.17400*, 2023. 1, 2, 3, 6

[7] Jeremy Cohen, Elan Rosenfeld, and Zico Kolter. Certified adversarial robustness via randomized smoothing. In *international conference on machine learning*, pages 1310–1320. PMLR, 2019. 2, 3

[8] Bradley Efron. Tweedie's formula and selection bias. *Journal of the American Statistical Association*, 106 (496):1602–1614, 2011. 3

[9] Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, et al. Scaling rectified flow transformers for high-resolution image synthesis. In *Forty-first international conference on machine learning*, 2024. 1, 2

[10] 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. 2

[11] Alhussein Fawzi, Seyed-Mohsen Moosavi-Dezfooli, and Pascal Frossard. Robustness of classifiers: from adversarial to random noise. *Advances in neural information processing systems*, 29, 2016. 3

[12] Pierre Foret, Ariel Kleiner, Hossein Mobahi, and Behnam Neyshabur. Sharpness-aware minimization for efficiently improving generalization. *arXiv preprint arXiv:2010.01412*, 2020. 2, 3, 6

[13] Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. In *International Conference on Learning Representations (ICLR)*, 2015. 2, 3

[14] Haoran He, Yuxiao Ye, Jie Liu, Jiajun Liang, Zhiyong Wang, Ziyang Yuan, Xintao Wang, Hangyu Mao, Pengfei Wan, and Ling Pan. Gardo: Reinforcing diffusion models without reward hacking. *arXiv preprint arXiv:2512.24138*, 2025. 2

[15] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022. 2

[16] Susung Hong. Smoothed energy guidance: Guiding diffusion models with reduced energy curvature of attention. *arXiv preprint arXiv:2408.00760*, 2024.

[17] Susung Hong, Gyuseong Lee, Wooseok Jang, and Seungryong Kim. Improving sample quality of diffusion models using self-attention guidance. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7462–7471, 2023. 2

[18] Borja Ibarz, Jan Leike, Tobias Pohlen, Geoffrey Irving, Shane Legg, and Dario Amodei. Reward learning from human preferences and demonstrations in atari. *Advances in neural information processing systems*, 31, 2018. 2

[19] Bahjat Kawar, Roy Ganz, and Michael Elad. Enhancing diffusion-based image synthesis with robust classifier guidance. *arXiv preprint arXiv:2208.08664*, 2022. 2

[20] Kwanyoung Kim and Byeongsu Sim. Pladis: Pushing the limits of attention in diffusion models at inference time by leveraging sparsity. *arXiv preprint arXiv:2503.07677*, 2025. 2

[21] Kwanyoung Kim and Jong Chul Ye. Noise2score: Tweedie's approach to self-supervised image denoising without clean images. *Advances in Neural Information Processing Systems*, 34:864–874, 2021. 3

[22] 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. 5, 6

[23] Black Forest Labs. Flux. https://github.com/black-forest-labs/flux, 2024. 6, 3

[24] Mathias Lecuyer, Vaggelis Atlidakis, Roxana Geambasu, Daniel Hsu, and Suman Jana. Certified robustness to adversarial examples with differential privacy. In *2019 IEEE symposium on security and privacy (SP)*, pages 656–672. IEEE, 2019. 2, 3

[25] Hyun Kyu Lee and Sung Whan Yoon. Flat reward in policy parameter space implies robust reinforcement learning. In *The Thirteenth International Conference on Learning Representations*, 2025. 2

[26] Taehwan Lee, Kyeongkook Seo, Jaejun Yoo, and Sung Whan Yoon. Understanding flatness in generative models: Its role and benefits. *arXiv preprint arXiv:2503.11078*, 2025. 2

[27] Bai Li, Changyou Chen, Wenlin Wang, and Lawrence Carin. Certified adversarial robustness with additive noise. *Advances in neural information processing systems*, 32, 2019. 3

[28] Binghui Li, Jikai Jin, Han Zhong, John Hopcroft, and Liwei Wang. Why robust generalization in deep learning is difficult: Perspective of expressive power. *Advances in Neural Information Processing Systems*, 35: 4370–4384, 2022. 2, 3

[29] 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 *The Thirty-ninth Annual Conference on Neural Information Processing Systems*. 1

[30] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. In *International Conference on Learning Representations (ICLR)*, 2018. 3

[31] Seyed-Mohsen Moosavi-Dezfooli, Alhussein Fawzi, and Pascal Frossard. Deepfool: a simple and accurate method to fool deep neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, pages 2574–2582, 2016. 2, 3

[32] Seyed-Mohsen Moosavi-Dezfooli, Alhussein Fawzi, Omar Fawzi, and Pascal Frossard. Universal adversarial perturbations. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, pages 1765–1773, 2017. 2, 3

[33] Preetum Nakkiran. Adversarial robustness may be at odds with simplicity. *arXiv preprint arXiv:1901.00532*, 2019. 2, 3

[34] 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. *arXiv preprint arXiv:2307.01952*, 2023. 1

[35] Mihir Prabhudesai, Anirudh Goyal, Deepak Pathak, and Katerina Fragkiadaki. Aligning text-to-image diffusion models with reward backpropagation. *openreview*, 2023. 1, 2, 3, 6

[36] Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. *Advances in neural information processing systems*, 36:53728–53741, 2023. 2, 8

[37] Aditi Raghunathan, Jacob Steinhardt, and Percy Liang. Certified defenses against adversarial examples. *arXiv preprint arXiv:1801.09344*, 2018. 3

[38] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10684–10695, 2022. 1, 2

[39] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L Denton, Kamyar Ghasemipour, Raphael Gontijo Lopes, Burcu Karagol Ayan, Tim Salimans, et al. Photorealistic text-to-image diffusion models with deep language understanding. *Advances in neural information processing systems*, 35:36479–36494, 2022. 6

[40] Hadi Salman, Jerry Li, Ilya Razenshteyn, Pengchuan Zhang, Huan Zhang, Sebastien Bubeck, and Greg Yang. Provably robust deep learning via adversarially trained smoothed classifiers. *Advances in neural information processing systems*, 32, 2019. 3

[41] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017. 2, 8

[42] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020. 3

[43] Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks. *arXiv preprint arXiv:1312.6199*, 2014. 2, 3

[44] Pascal Vincent. A connection between score matching and denoising autoencoders. *Neural computation*, 23 (7):1661–1674, 2011. 3

[45] 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*, pages 8228–8238, 2024. 2

[46] Yibin Wang, Zhimin Li, Yuhang Zang, Yujie Zhou, Jiazi Bu, Chunyu Wang, Qinglin Lu, Cheng Jin, and Jiaqi Wang. Pref-grpo: Pairwise preference reward-based grpo for stable text-to-image reinforcement learning. *arXiv preprint arXiv:2508.20751*, 2025. 2

[47] Dongxian Wu, Shu-Tao Xia, and Yisen Wang. Adversarial weight perturbation helps robust generalization. In *Advances in Neural Information Processing Systems (NeurIPS) 33*, 2020. 4

[48] Jie Wu, Yu Gao, Zilyu Ye, Ming Li, Liang Li, Hanzhong Guo, Jie Liu, Zeyue Xue, Xiaoxia Hou, Wei Liu, et al. Rewarddance: Reward scaling in visual generation. *arXiv preprint arXiv:2509.08826*, 2025. 1

[49] 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. 6

[50] Xiaoshi Wu, Yiming Hao, Manyuan Zhang, Keqiang Sun, Zhaoyang Huang, Guanglu Song, Yu Liu, and Hongsheng Li. Deep reward supervisions for tuning text-to-image diffusion models. In *European Conference on Computer Vision*, pages 108–124. Springer, 2024. 1, 2, 3, 6

[51] Enze Xie, Junsong Chen, Junyu Chen, Han Cai, Haotian Tang, Yujun Lin, Zhekai Zhang, Muyang Li, Ligeng Zhu, Yao Lu, et al. Sana: Efficient high-resolution image synthesis with linear diffusion transformers. *arXiv preprint arXiv:2410.10629*, 2024. 2

[52] 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. 1, 2, 3, 6

[53] 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, 2024. 5, 6

[54] 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*, pages 8941–8951, 2024. 2

[55] Chaojian Yu, Bo Han, Mingming Gong, Li Shen, Shiming Ge, Bo Du, and Tongliang Liu. Robust weight perturbation for adversarial training. *arXiv preprint arXiv:2205.14826*, 2022. 4

[56] Hongyang Zhang, Yaodong Yu, Jiantao Jiao, Eric Xing, Laurent El Ghaoui, and Michael Jordan. Theoretically principled trade-off between robustness and accuracy. In *International conference on machine learning*, pages 7472–7482. PMLR, 2019. 3

[57] Yihao Zhang, Hangzhou He, Jingyu Zhu, Huanran Chen, Yifei Wang, and Zeming Wei. On the duality between sharpness-aware minimization and adversarial training. *arXiv preprint arXiv:2402.15152*, 2024. 2, 4

[58] Huaisheng Zhu, Teng Xiao, and Vasant G Honavar. DSPO: Direct score preference optimization for diffusion model alignment. In *The Thirteenth International Conference on Learning Representations*, 2025. 2