MPO: Single-Stream Policy Optimization for Efficient Text-to-Image Alignment

Lishuai Gao[1,2], Jie Hu[2], Cong Wei[2]*, Yujie Zhong[2], Yibo Zhao[1], Zan Gao[1]**, and Xiaoming Wei[2]

[1] Tianjin University of Technology
[2] Meituan Inc.

Abstract. Online reinforcement learning (RL) has become an effective tool for aligning text-to-image models with human preferences, but its efficiency is often limited by group-relative variance reduction. Existing methods generate multiple trajectories per prompt to estimate local baselines, which introduces synchronization overhead and can weaken the learning signal when group rewards become homogeneous. We propose Monolithic Policy Optimization (MPO), a group-free online RL framework based on a single-stream training principle: one prompt, one trajectory, and one policy update. MPO retains stochastic differential equation (SDE) exploration within each trajectory, while replacing group-wise baselines with a persistent Bayesian value tracker. The tracker maintains a history-aware reward estimate for each prompt and adapts its uncertainty using a Girsanov-inspired policy-drift proxy, providing stable global advantage estimation under non-stationary policy updates. Across compositional generation, visual text rendering, and human-preference alignment benchmarks, MPO consistently improves alignment quality over group-based baselines. It also delivers a 26× wall-clock speedup and a 5× sample-efficiency gain under matched training settings.

Keywords: Text-to-Image Alignment · Online Reinforcement Learning · Bayesian Value Tracking · Variance Reduction

1 Introduction

Text-to-image models based on diffusion and flow-matching principles [15,23,35] have enabled high-quality visual generation, yet aligning them with human preferences remains a central challenge. Practical alignment objectives, including aesthetic quality, photorealism, visual text rendering, and compositional instruction following, are often non-differentiable and therefore naturally suited to reinforcement learning (RL) [30,52]. Recent online RL methods for visual generation, referred to here as the GRPO series [22, 24, 49], typically adopt a group-relative design: for each prompt, multiple images are sampled in parallel and compared

* Corresponding author.
** Corresponding author.

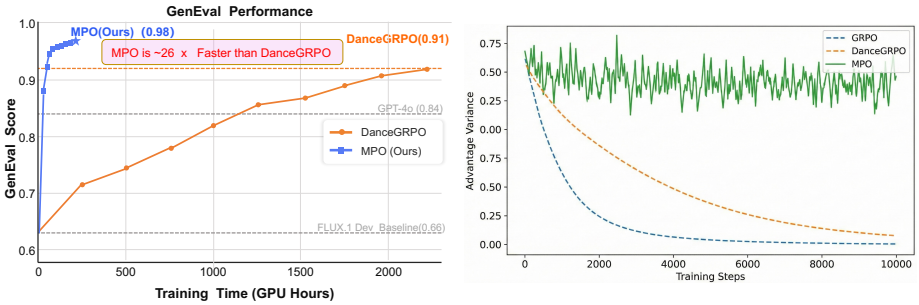

Fig. 1: The advantage of Monolithic Policy Optimization. Left: MPO reaches the target GenEval score with substantially lower wall-clock cost than DanceGRPO. Right: group-relative training suffers from rapidly diminishing intra-group advantage variance, whereas MPO maintains a stable global learning signal throughout training.

within the group to estimate a local advantage baseline, as illustrated in Figure 2(a).

Despite their effectiveness, group-relative methods introduce two practical limitations. First, parallel group sampling creates synchronization overhead, since an update can only proceed after all trajectories in the group have finished generation. This effect becomes more pronounced for large flow-based models and multi-step stochastic samplers. Second, as the policy improves, samples generated from the same prompt often receive increasingly similar rewards. The resulting reduction in intra-group reward variance weakens the advantage signal and reduces the utility of expensive parallel samples. These limitations motivate a simpler question: can online RL for text-to-image alignment retain stable variance reduction without relying on concurrent groups?

We answer this question with Monolithic Policy Optimization (MPO), a group-free online RL framework built on a single-stream principle: one prompt $\rightarrow$ one trajectory $\rightarrow$ one policy update. As shown in Figure 2(b), MPO removes the need for group synchronization and allows each generated trajectory to contribute directly to training. The main challenge is that single-trajectory updates no longer have access to an on-the-fly group baseline. MPO addresses this by combining stochastic differential equation (SDE) exploration with a persistent Bayesian value tracker. For each prompt, the tracker maintains a history-aware posterior estimate of expected reward, whose uncertainty is updated according to a Girsanov-inspired policy-drift proxy. This adaptive uncertainty prevents the baseline from becoming stale under non-stationary policy updates. Together with global advantage normalization and an uncertainty-guided curriculum, MPO provides stable learning signals while preserving the computational simplicity of single-stream training.

Our contributions are summarized as follows:

1. We analyze the efficiency and learning-signal limitations of group-relative online RL for text-to-image alignment, focusing on synchronization overhead and intra-group advantage collapse.

2. We propose Monolithic Policy Optimization (MPO), a group-free single-stream RL framework that improves training efficiency without requiring parallel samples per prompt.
3. We introduce a Bayesian value tracker with policy-drift-aware uncertainty adaptation, which replaces local group baselines with stable global advantage estimation in non-stationary online training.

Experiments across compositional generation, visual text rendering, and human-preference alignment show that MPO consistently improves alignment quality over group-based baselines. Under matched training settings, MPO achieves a 26× wall-clock speedup and a 5× gain in sample efficiency, while maintaining strong performance across multiple foundational architectures.

2 Related Work

2.1 Alignment of Text-to-Image Generative Models

Aligning text-to-image (T2I) models with human preferences has become an important step for improving perceptual quality, prompt adherence, and compositional controllability. Existing alignment methods can be broadly grouped into three lines. Reward-Weighted Regression (RWR) treats alignment as supervised fine-tuning on high-reward samples [6, 21, 31]. Direct Preference Optimization (DPO) and its variants optimize preference pairs without explicitly learning a reward-conditioned policy gradient [34, 41]. These methods are often sample-efficient, but their performance depends heavily on the coverage and quality of preference data. A third line applies explicit reinforcement learning (RL) to optimize non-differentiable reward signals for generative policies [4,8]. Such methods, commonly built on PPO-style objectives [39], provide direct control over continuous reward optimization, but also introduce challenges in variance reduction, training stability, and computational cost. MPO belongs to this RL-based family and focuses on improving the efficiency and stability of online policy optimization.

2.2 Efficient Policy Optimization for Diffusion Models

Applying RL to diffusion and flow-matching models often requires stochastic sampling, typically by converting deterministic ODE sampling into SDE-based exploration [4]. Recent GRPO-based methods [24, 49] reduce gradient variance by generating multiple samples for each prompt and estimating a group-relative baseline. This design has proven effective, but its efficiency is limited by group synchronization and by the collapse of intra-group reward variance as samples become similar. MixGRPO [22] improves this paradigm through dual-mode sampling, using ODE generation for efficiency and SDE sampling for intra-group diversity, while still relying on group-wise advantage estimation.

Other works seek to reduce or avoid group sampling. DiffusionNFT [51] eliminates group construction by formulating alignment as reward-weighted velocity

regression on the forward process with deterministic ODE solvers. This provides strong efficiency, but changes the optimization route away from the standard reverse-time stochastic generation process. Gradient-based or direct-alignment methods such as Direct-Align and SRPO [9] further reduce generation cost by bypassing online RL sampling, but they often require reward gradients, preference pairs, or architecture-specific assumptions. In contrast, MPO keeps the reverse-time SDE trajectory used during online RL, removes group-relative sampling, and replaces local group baselines with a persistent Bayesian value tracker compatible with black-box reward models.

2.3 Reward Models, Over-optimization, and Policy Drift

The effectiveness of RL alignment depends strongly on the reward model used as the training signal. Common reward models, including HPSv2 [45], PickScore [19], and ImageReward [48], are trained from large-scale human preference data and often rely on vision-language backbones such as CLIP [33]. MPO treats these reward models as black-box evaluators and does not require reward model fine-tuning.

A related challenge is reward over-optimization, where the policy exploits imperfections of the proxy reward model and degrades perceptual quality or prompt faithfulness [10, 27]. Existing mitigation strategies include KL regularization against the reference model [24,39] and early stopping [4]. Online RL also faces policy drift: as the policy changes, the expected reward associated with a prompt becomes non-stationary. Bayesian filters such as Kalman Temporal Differences have been used to track non-stationary value estimates in traditional RL [11]. MPO adapts this idea to T2I alignment by maintaining a prompt-wise Bayesian reward estimate and updating its uncertainty with a Girsanov-inspired policy-drift proxy, yielding a history-aware baseline for stable global advantage estimation.

3 Method

Preliminary. We formulate flow-based text-to-image generation as a reinforcement learning problem. Given a prompt c, a flow-matching model defines a continuous-time policy through its drift field $\mathbf{v}\theta(\mathbf{z}t, c, t)$ over $t \in [0, 1]$. Starting from an initial latent $\mathbf{z}0 \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$, the model produces a terminal latent $\mathbf{z}1$ and an image $\mathbf{x} = \mathrm{Dec}(\mathbf{z}1)$, where $\mathrm{Dec}(\cdot)$ denotes the VAE decoder [18]. We view the denoising path $\tau = \mathbf{z}0, \mathbf{z}t_1, \ldots, \mathbf{z}1$ as a rollout of a Markov Decision Process [4], where the state is $s_t = (\mathbf{z}t, c, t)$, the action corresponds to the latent increment induced by the drift, and the terminal reward is $r = R\phi(\mathbf{x}, c)$. The alignment objective is

$$\mathcal{J}(\theta) = \mathbb{E}\mathbf{z}0 \sim \mathcal{N}(\mathbf{0}, \mathbf{I}), , c \sim \mathcal{D}\left[R_\phi\left(\mathrm{Dec}\left(\mathbf{z}1(\mathbf{z}0, c, \theta)\right), c\right)\right]. \tag{1}$$

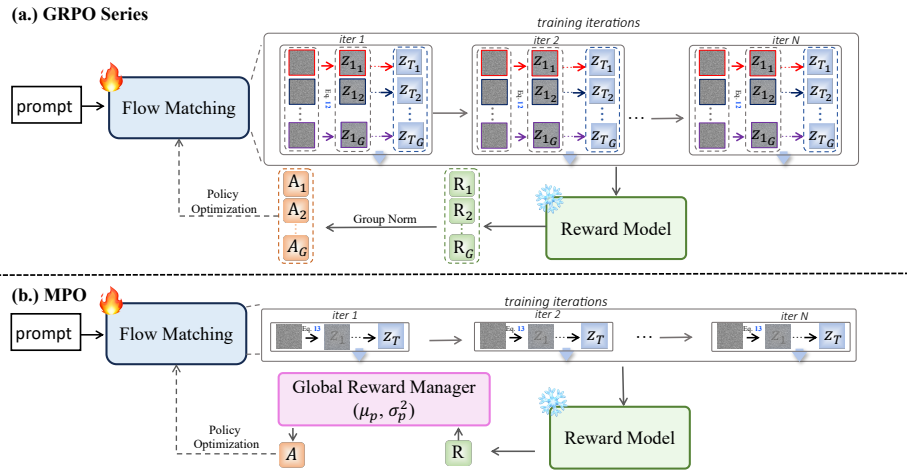

Fig. 2: MPO vs. the GRPO-Series. (a) Group-relative methods such as Flow-GRPO sample G trajectories for each prompt and estimate an on-the-fly local baseline. This design introduces synchronization overhead because each update waits for the slowest trajectory, and its learning signal can collapse when intra-group rewards become nearly identical. (b) MPO removes the group abstraction and follows a single-stream principle: one prompt, one trajectory, and one policy update. The missing group baseline is replaced by a global reward manager, which maintains a Bayesian value tracker for each prompt and adapts its uncertainty using a Girsanov-inspired drift proxy.

3.1 Monolithic Policy Optimization

Existing online RL methods for diffusion or flow models, including the GRPO series [22, 24, 49], reduce variance by sampling multiple trajectories for the same prompt. Although this local comparison is simple, it couples the update to a group of generations. As a result, the update latency is determined by the slowest sample in the group, and the advantage signal weakens when the sampled images receive similar rewards.

MPO removes this group dependency. For each prompt, it samples one stochastic trajectory and immediately extracts a policy update from that trajectory. This single-stream design avoids group synchronization, but it also requires a baseline that remains stable without concurrent samples. MPO addresses this with three components: stochastic trajectory exploration, a persistent Bayesian value tracker, and global advantage normalization.

3.2 Single-Trajectory Exploration and Drift Estimation

Since MPO does not rely on multiple initial latents within a group, exploration must be introduced along the trajectory itself. We therefore sample the generative path with an SDE [40]:

$$\mathrm{d}\mathbf{z} = \mathbf{v}\theta(\mathbf{z}t, c, t), \mathrm{d}t + g(t), \mathrm{d}\mathbf{w}, \tag{2}$$

where d$\mathbf{w}$ is the Wiener noise and $g(t)$ controls the diffusion strength. In practice, we use the Euler-Maruyama discretization

$$\mathbf{z}t + \Delta t = \mathbf{z}t + \mathbf{v}\theta(\mathbf{z}t, c, t)\Delta t + g_t\sqrt{\Delta t}, \varepsilon t, \qquad \varepsilon t \sim \mathcal{N}(\mathbf{0}, \mathbf{I}). \tag{3}$$

During training, the initial latent $\mathbf{z}_0$ associated with a prompt is fixed. This design isolates reward changes caused by the policy update from those caused by resampling the initial noise, making the historical reward tracker more stable.

The value tracker must also account for policy drift. Let θ_k and θ_{k-1} denote the current and previous policies at training iteration k. For SDEs with shared diffusion coefficient, Girsanov's theorem [28] motivates measuring drift changes through the path-space divergence

$$D_{\mathrm{KL}}\left(\mathbb{P}\theta_k, |, \mathbb{P}\theta_{k-1}\right) = \mathbb{E}\mathbb{P}\theta_k \left[\int_0^1 \frac{|\mathbf{v}\theta_k(\mathbf{z}t, c, t) - \mathbf{v}\theta k - 1(\mathbf{z}t, c, t)|\, 2^2}{2g(t)^2}, \mathrm{d}t\right]. \tag{4}$$

Computing this quantity exactly during online training is expensive and sensitive to small values of $g(t)$. We therefore use a tractable drift proxy computed on the sampled trajectory:

$$\widehat{D}\mathrm{drift}(\theta_k, \theta k - 1) = \mathbb{E}t, , \mathbf{z}t \sim \tau \left[|\mathbf{v}\theta_k(\mathbf{z}t, c, t) - \mathbf{v}\theta k - 1(\mathbf{z}t, c, t)|\, 2^2\right]. \tag{5}$$

This proxy is not an unbiased estimator of the path-space KL in Eq. (14). We use it only as an empirical indicator of policy movement for adaptive value tracking.

3.3 Bayesian Value Tracker

For each prompt c, MPO maintains a Gaussian posterior over its expected reward,

$$\mathcal{V}(c) = \mathcal{N}(\mu_c, \sigma_c^2), \tag{6}$$

where μ_c serves as the baseline and σ_c^2 represents the uncertainty of the estimate. After observing a reward r at training iteration k, the tracker is updated with a Kalman-style rule [17]:

$$K_k = \frac{\sigma_{c,k-1}^2}{\sigma_{c,k-1}^2 + \sigma_{\mathrm{obs}}^2},\ \mu_{c,k} \quad \leftarrow \mu_{c,k-1} + K_k\left(r - \mu_{c,k-1}\right),\ \sigma_{c,k}^2 \leftarrow (1 - K_k)\,\sigma_{c,k-1}^2 + Q_k. \tag{7}$$

Here K_k is the Kalman gain, and σ_{obs}^2 denotes the observation noise of the reward model. In online RL, the expected reward for a prompt changes as the policy changes. If the process noise is set too small, the tracker becomes over-confident and cannot adapt to new rewards. MPO therefore sets

$$Q_k = \alpha, \widehat{D}\mathrm{drift}(\theta_k, \theta k - 1), \tag{8}$$

where α is a scaling coefficient. Larger policy changes increase the posterior uncertainty and allow the tracker to rely more on recent observations, while small updates preserve the benefit of historical smoothing.

3.4 Global Advantage Normalization and Prompt Curriculum

Given the tracked baseline, the raw advantage for prompt c is

$$A = r - \mu_{c,k}. \tag{9}$$

Instead of normalizing advantages within a prompt-level group, MPO normalizes them globally across recent batches. We maintain exponential moving averages of the first and second moments:

$$\mu_A \leftarrow \lambda\mu_A + (1-\lambda)A, \qquad \sigma_A^2 \leftarrow \lambda\sigma_A^2 + (1-\lambda)(A-\mu_A)^2. \tag{10}$$

The normalized advantage is

$$\tilde{A} = \frac{A - \mu_A}{\sqrt{\sigma_A^2 + \varepsilon}}. \tag{11}$$

This global normalization prevents the learning signal from vanishing when samples for a prompt become similar.

The posterior uncertainty also defines an online prompt curriculum. Inspired by UCB-style exploration [1, 14], MPO samples prompt c with probability

$$p(c) \propto \sigma_c + \frac{\eta}{\sqrt{n_c + 1}}, \tag{12}$$

where n_c is the number of previous visits to prompt c, and η controls the strength of coverage regularization. The uncertainty term prioritizes prompts whose reward estimates remain unreliable, while the visit-count term prevents rarely sampled prompts from being ignored [2].

3.5 Advantage-Weighted Policy Update

The terminal reward is non-differentiable with respect to the generated image. Rather than using a high-variance score-function estimator such as REINFORCE [44], MPO adopts an advantage-weighted regression objective [31]. For a sampled trajectory τ, let $\mathbf{u}\tau(\mathbf{z}t, \mathbf{z}_0)$ denote the empirical target vector field associated with the realized path. In rectified-flow models, this target corresponds to the vector that reconstructs the sampled trajectory.

To emphasize informative samples, we compute the normalized reward surprise

$$S_c = \frac{|r - \mu_{c,k}|}{\sigma_{c,k} + \varepsilon} \tag{13}$$

and set $w_c = 1 + \gamma S_c$, following the principle of prioritized updates [37]. The MPO loss is

$$\mathcal{L}\mathrm{MPO}(\theta) = \mathbb{E}t, , \mathbf{z}t \sim \tau \left[\mathrm{sg}\left(w_c \tilde{A} \right) |\mathbf{v}\theta_k(\mathbf{z}t, c, t) - \mathbf{u}\tau(\mathbf{z}t, \mathbf{z}0)|\, 2^2 \right], \tag{14}$$

where $\mathrm{sg}(\cdot)$ denotes the stop-gradient operation. The target $\mathbf{u}\tau$ is computed from the current sampled trajectory rather than from the pretrained base model. A

positive normalized advantage increases the likelihood of trajectories that lead to high rewards, while a negative advantage suppresses low-reward trajectories. In this way, MPO obtains a stable single-stream policy update without constructing prompt-level sample groups.

4 Experiments

We evaluate MPO from four aspects. First, we test whether the proposed single-stream update improves compositional text-to-image alignment on GenEval. Second, we examine visual text rendering and human-preference alignment to verify that the method is not specialized to one reward. Third, we measure wall-clock efficiency under matched hardware and training settings. Finally, we conduct ablations and reviewer-requested validations to analyze the sources of the gain and the stability of the method.

4.1 Experimental Setup

Implementation Details. Our main experiments are conducted on FLUX.1 Dev [20], a representative flow-matching text-to-image model. To examine whether the proposed optimization is architecture-specific, we additionally evaluate MPO on Stable Diffusion 3.5-Medium (SD3.5-M) [7] and Stable Diffusion v1.5. Unless otherwise stated, all controlled comparisons use the same base model, prompt set, reward model, optimizer, batch size, and sampler configuration. We optimize the policy with AdamW [25], using a constant learning rate of `5e-6` and a global batch size of `64`. Trajectories are generated with a 12-step Euler-Maruyama SDE sampler. The diffusion coefficient g_t is linearly annealed from `0.1` to `0`. The EMA decay for global advantage normalization is $\lambda = 0.99$, the curriculum coefficient is $\eta = 1.0$, and the reward-surprise weight is $\gamma = 0.5$. All efficiency measurements are obtained on NVIDIA H800 GPUs.

Data Protocol. To reduce train-test leakage, we decontaminate the training prompts against the evaluation prompts. Specifically, we compute the cosine similarity between CLIP text embeddings and remove any training prompt whose similarity to an evaluation prompt is at least 0.8. Key quantitative results are averaged over three random seeds, with standard deviations below 0.01 on the main metrics. During training, the initial latent $\mathbf{z}_0$ associated with each prompt is fixed. This is used only to stabilize online value tracking by isolating reward changes caused by policy updates from those caused by resampling the initial noise. During inference and evaluation, random seeds are not restricted.

Baselines. Our primary controlled baseline is DanceGRPO [49], a strong group-relative online RL method. For SD3.5-M, we also compare against Flow-GRPO [24], an efficiency-oriented group-based baseline. In the broader GenEval comparison, we include representative autoregressive, diffusion, flow-matching, and RL-aligned models reported in prior work. For controlled comparisons, the same optimizer, sampler, base model, and reward are used whenever applicable.

Table 1: GenEval Result. We **highlight** the best scores. Obj.: Object; Attr.: Attribution. † indicates that, to ensure a fair comparison of MPO effectiveness, we train FLUX.1 Dev using the same training dataset.

Model	Overall	Single Obj.	Two Obj.	Counting	Colors	Position	Attr. Binding
Autoregressive Models:							
Show-o [47]	0.53	0.95	0.52	0.49	0.82	0.11	0.28
Emu3-Gen [42]	0.54	0.98	0.71	0.34	0.81	0.17	0.21
JanusFlow [26]	0.63	0.97	0.59	0.45	0.83	0.53	0.42
Janus-Pro-7B [5]	0.80	0.99	0.89	0.59	0.90	0.79	0.66
GPT-4o [16]	0.84	0.99	0.92	0.85	0.92	0.75	0.61
Diffusion Models:							
LDM [35]	0.37	0.92	0.29	0.23	0.70	0.02	0.05
SD1.5 [35]	0.43	0.97	0.38	0.35	0.76	0.04	0.06
SD2.1 [35]	0.50	0.98	0.51	0.44	0.85	0.07	0.17
SD-XL [32]	0.55	0.98	0.74	0.39	0.85	0.15	0.23
DALLE-2 [29]	0.52	0.94	0.66	0.49	0.77	0.10	0.19
DALLE-3 [3]	0.67	0.96	0.87	0.47	0.83	0.43	0.45
Flow Matching Models:							
SD3.5-M [7]	0.63	0.98	0.78	0.50	0.81	0.24	0.52
SD3.5-L [7]	0.71	0.98	0.89	0.73	0.83	0.34	0.47
SANA-1.5 4.8B [46]	0.81	0.99	0.93	0.86	0.84	0.59	0.65
FLUX.1 Dev [20]	0.66	0.98	0.81	0.74	0.79	0.22	0.45
Flow-GRPO [24]	0.95	**1.00**	0.99	0.95	0.92	**0.99**	0.86
FLUX.1 Dev + DanceGRPO †	0.91	0.93	0.94	0.96	0.88	0.91	0.87
FLUX.1 Dev + MPO	**0.98**	**1.00**	**1.00**	**0.99**	**0.94**	**0.99**	**0.93**

Table 2: Performance on Compositional Image Generation, Visual Text Rendering, and Human Preference benchmarks, evaluated by task performance on test prompts, and by image quality and preference scores on DrawBench prompts. ImgRwd: ImageReward; UniRwd: UnifiedReward.

Model	Task Metric			Image Quality		Preference Score		
	GenEval	OCR	PickScore	Aesthetic	DeQA	ImgRwd	PickScore	UniRwd
FLUX.1 Dev	0.66	0.61	21.75	5.51	4.04	0.86	21.66	3.35
Compositional Image Generation:								
DanceGRPO	0.91	—	—	5.23	4.04	1.01	21.95	3.47
MPO	0.98	—	—	5.49	4.06	1.16	23.17	3.75
Visual Text Rendering:								
DanceGRPO	—	0.91	—	5.29	4.06	0.99	20.15	3.37
MPO	—	0.97	—	5.53	4.14	1.15	23.47	3.68
Human Preference Alignment:								
DanceGRPO	—	—	23.28	5.86	4.23	1.22	23.42	3.59
MPO	—	—	23.89	6.08	4.40	1.32	23.91	3.74

Evaluation Metrics. We use GenEval [12] to measure compositional generation, including object presence, counting, color, position, and attribute binding. For visual text rendering, we follow the normalized edit-distance metric $(1-N_e/N_{\text{ref}})$ used in [13]. For human-preference alignment, we report PickScore [19]. To monitor potential reward over-optimization, we also evaluate out-of-domain image-quality and preference metrics on DrawBench [36], including Aesthetic Score [38], DeQA [50], ImageReward [48], and UnifiedReward [43].

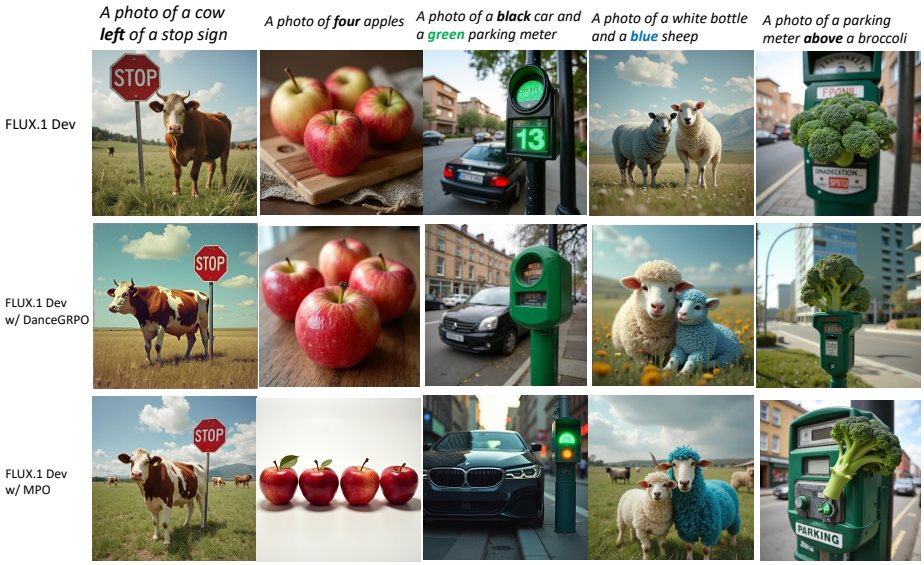

Fig. 3: Qualitative comparison of text-to-image alignment. Images for each prompt are generated with the same initial latent noise $\mathbf{z}_0$ to isolate the effect of policy optimization. MPO generally improves instruction adherence while preserving the visual quality of the base model. Some hard cases remain, especially for fine-grained attribute binding and uncommon spatial relations.

4.2 Main Results

Compositional Image Generation. The GenEval comparison table reports the main compositional results. MPO reaches an overall score of **0.98**, improving over the base FLUX.1 Dev model, DanceGRPO, and Flow-GRPO. The improvement is especially clear on categories that require precise semantic control, including counting, position, and attribute binding. These results show that removing prompt-level groups does not weaken alignment. Instead, when single-trajectory updates are paired with a persistent value tracker and global advantage normalization, each generated sample can contribute a useful learning signal.

Visual Text Rendering and Preference Alignment. The multi-task benchmark table extends the evaluation to visual text rendering and human-preference optimization. On FLUX.1 Dev, MPO improves OCR accuracy from 0.91 to **0.97** over DanceGRPO and improves PickScore from 23.28 to **23.89**. The out-of-domain metrics on DrawBench are also preserved or improved, including Aesthetic Score, DeQA, ImageReward, PickScore, and UnifiedReward. This is important because optimizing a single reward can easily degrade general image quality. The results indicate that MPO improves alignment without introducing obvious reward over-optimization on independent quality and preference measures.

Table 3: Performance on Compositional Image Generation, Visual Text Rendering, and Human Preference benchmarks, evaluated by task performance on test prompts, and by image quality and preference scores on DrawBench prompts. ImgRwd: ImageReward; UniRwd: UnifiedReward.

Model	Task Metric			Image Quality		Preference Score		
	GenEval	OCR	PickScore	Aesthetic	DeQA	ImgRwd	PickScore	UniRwd
SD3.5-M	0.63	0.59	21.72	5.39	4.07	0.87	22.34	3.33
Compositional Image Generation:								
Flow-GRPO	0.95	—	—	5.25	4.01	1.03	22.37	3.51
MPO	0.98	—	—	5.39	4.08	1.14	23.22	3.69
Visual Text Rendering:								
Flow-GRPO	—	0.92	—	5.32	4.06	0.95	22.44	3.42
MPO	—	0.97	—	5.49	4.12	1.08	23.01	3.56
Human Preference Alignment:								
Flow-GRPO	—	—	23.31	5.92	4.22	1.28	23.53	3.66
MPO	—	—	23.91	6.12	4.43	1.35	24.02	3.77

Qualitative Analysis. Fig. 3 compares FLUX.1 Dev, DanceGRPO, and MPO under matched initial latents. The base model often misses object counts or spatial relations. DanceGRPO improves instruction following, but can introduce over-saturated colors or local artifacts. MPO better preserves the base model's photorealistic appearance while improving compositional control. At the same time, the qualitative results should not be read as eliminating all failures. Difficult cases remain for unusual object relations and fine-grained attribute binding, such as prompts involving rare spatial configurations. We therefore include a blind human preference study in Table 5 to evaluate whether the qualitative trend is systematic rather than cherry-picked.

Generality Across Architectures. The SD3.5-M comparison table shows that MPO also improves over the SD3.5-M base model and Flow-GRPO across GenEval, visual text rendering, PickScore, and out-of-domain quality metrics. We further evaluate the U-Net-based Stable Diffusion v1.5. MPO improves its GenEval score from 0.43 to 0.55 and PickScore from 19.5 to 21.5. These results suggest that the gain comes from the optimization design rather than from a FLUX-specific architectural property.

Robustness to Multiple Rewards. Real deployment often requires optimizing more than one preference signal, such as preserving image quality while enforcing accurate text rendering. We therefore evaluate a scalarized reward combining PickScore and OCR accuracy. In this setting, group-relative methods are more likely to suffer from unstable or collapsed local advantages because the reward variance can increase across objectives. MPO remains stable because its value tracker smooths reward estimates over time and its global normalization keeps the gradient scale controlled. Under the dual-reward setting, MPO reaches 0.82

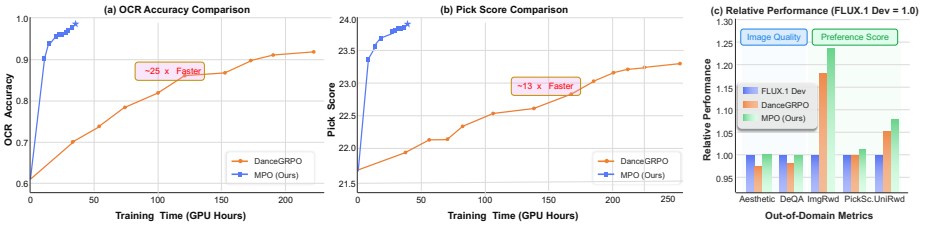

Fig. 4: Efficiency analysis. (a)–(b) Wall-clock training curves show that MPO reaches target OCR and PickScore levels substantially faster than DanceGRPO. (c) Relative performance on DrawBench, normalized by FLUX.1 Dev, shows that MPO improves alignment without degrading out-of-domain quality metrics.

Table 4: Detailed efficiency comparison between MPO and DanceGRPO.

Metric	**MPO (Ours)**	**DanceGRPO**
Batch Size	64	64
Wall-clock Time / Iteration (s)	**0.85**	14.12
Peak GPU Memory (GB, per GPU)	60	68
Time to reach GenEval Score = 0.91		
Iterations Required	∼940	∼6,700
Total GPU Hours	**∼9**	**∼235**
Speedup Factor	**235/9 = 26×**	

OCR accuracy while preserving PickScore at 21.5, improving over the corresponding group-based baseline.

4.3 Efficiency Analysis

Wall-Clock Speed. Fig. 4 and the efficiency table summarize the training cost. A central benefit of MPO is that it removes group-level synchronization. In DanceGRPO with group size $G = 8$, an update must wait for the slowest trajectory in the group. Under the matched FLUX.1 Dev setting, the per-sample generation time within DanceGRPO groups has mean 12.3s, maximum 18.7s, and standard deviation 3.2s. This creates a 52% wait-time overhead relative to the mean generation time. MPO removes this barrier by using one trajectory per prompt and one immediate update.

Sample Efficiency. MPO also requires fewer generated samples to reach the same target score. The persistent Bayesian tracker allows each single trajectory to be compared against a history-aware baseline, rather than against only the other samples in the same group. This avoids wasting full groups whose rewards are nearly identical. In the matched FLUX.1 Dev comparison, MPO reaches the target GenEval score using about 7.1× fewer iterations and about 5× fewer generated samples than DanceGRPO.

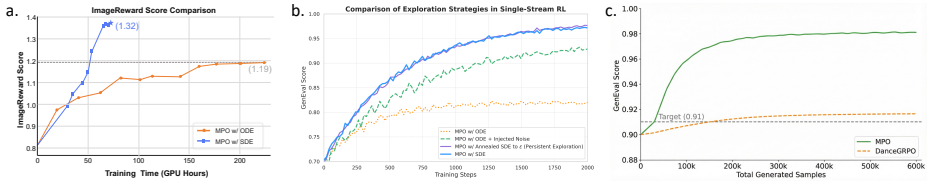

Fig. 5: Exploration and sample efficiency. (a) The deterministic ODE variant stagnates on ImageReward, showing the need for trajectory-level exploration. (b) SDE exploration outperforms simple injected noise on GenEval. (c) With stable exploration and global normalization, MPO reaches the target score with about 5× fewer generated samples than DanceGRPO.

Compound Acceleration. The wall-clock gain comes from both statistical and structural factors. Statistically, MPO needs fewer updates because the value tracker extracts a usable advantage from each sample. Structurally, each update is faster because no group synchronization is required. The measured update time decreases from 14.12s for DanceGRPO to 0.85s for MPO. As a result, MPO reaches the target performance in about 9 GPU hours, compared with 235 GPU hours for DanceGRPO, giving a measured 26× wall-clock speedup under matched H800 hardware.

4.4 Ablation Study

Necessity of SDE Exploration. Single-stream RL no longer obtains diversity from multiple group samples. We therefore test whether trajectory-level stochasticity is necessary. As shown in Fig. 5(a), the deterministic ODE variant learns initially but soon stagnates. Fig. 5(b) further shows that simply injecting isotropic Gaussian noise into an ODE path is inferior to the SDE sampler. This suggests that the path-dependent stochasticity of the SDE provides more useful exploration than post-hoc noise injection. Fig. 5(c) connects this exploration benefit to sample efficiency: with SDE exploration and global normalization, MPO reaches the target alignment score with about 5× fewer generated samples.

Component Contributions. We ablate the main stabilizing components on GenEval. Removing global advantage normalization reduces the score from 0.98 to 0.93, showing that raw single-trajectory advantages are unstable. Disabling adaptive forgetting by setting $Q_k = 0$ reduces the score to 0.94, indicating that the value tracker must remain responsive to policy drift. Replacing the Bayesian tracker with a simple EMA baseline reduces the score to 0.96, and removing the uncertainty-guided curriculum also yields 0.96. These results show that the SDE sampler, adaptive value tracking, global normalization, and prompt curriculum are complementary. The single-stream design is therefore not merely a cheaper version of group-based training; it requires the proposed stabilization components to work reliably.

Table 5: Additional validations. We include the checks most relevant to stability and reviewer concerns: long-horizon behavior, multi-objective robustness, the role of fixed training latents, and blind human preference evaluation.

Study	Setting	Baseline / Variant	MPO
Extended training	GenEval at 2k iterations	DiffusionNFT: 0.53	**0.98**
Multi-objective stress	GenEval with PickScore + OCR	DiffusionNFT: 0.94	**0.98**
Fixed $\mathbf{z}_0$ ablation	GenEval / convergence time	Random $\mathbf{z}_0$: 0.93 / 19.3h	**0.98** / **9.1h**
Human preference	100 prompts, 10 raters	DanceGRPO win: 15%	MPO win: **58%**

Long-Horizon and Multi-Objective Stability. Table 5 integrates the additional checks most relevant to the reviewer concerns. First, although DiffusionNFT can reach a high peak GenEval score, its score drops under extended training, whereas MPO remains stable at 2k iterations. This suggests that preserving the reverse-time stochastic trajectory and using an adaptive value tracker improves late-stage stability. Second, under a multi-objective reward combining PickScore and OCR, MPO preserves its GenEval score better than the forward-process baseline. This supports the claim that MPO is not only efficient, but also robust when the reward contains competing factors.

Effect of Fixed Training Latents. Table 5 also shows that fixing $\mathbf{z}_0$ during training is important for stable value tracking. When $\mathbf{z}_0$ is randomized for the same prompt, the tracker must explain reward variation from both policy changes and latent changes. This increases tracker variance, slows convergence, and reduces GenEval from 0.98 to 0.93. Fixing $\mathbf{z}_0$ during training avoids this confound. Importantly, this does not restrict evaluation diversity, because random seeds are used during inference.

Human Preference Evaluation. To complement automatic metrics and address qualitative ambiguity, we conduct a blind human preference evaluation with 10 raters on 100 complex prompts. Each comparison uses matched prompts and matched initial latents. MPO wins 58

Hyperparameter Sensitivity. We evaluate the sensitivity of MPO to the adaptive scaling factor α, the curriculum coefficient η, and the surprise weight γ. MPO remains near its best GenEval performance for $\alpha \in [0.1, 5.0]$, indicating that the tracker is not brittle to the exact scale of the drift proxy. Very small η over-focuses training on a few uncertain prompts, while very large η approaches uniform sampling and weakens the curriculum. For reward-surprise weighting, $\gamma = 0.5$ provides the best trade-off between emphasizing informative samples and avoiding unstable updates. Overall, MPO remains stable across a broad range of hyperparameters and does not rely on a narrow tuning window.

5 Conclusion

In this work, we identified the prevalent group-relative paradigm as a structural bottleneck in online RL for text-to-image alignment, inherently susceptible to synchronization latency and gradient signal collapse. To resolve these fundamental flaws, we introduced Monolithic Policy Optimization (MPO), a strictly group-free, single-stream alignment framework. By decoupling variance reduction from intra-group sampling, MPO leverages a persistent Bayesian value tracker equipped with a Girsanov-derived adaptive forgetting mechanism. This formulation successfully tames the high variance of intrinsic SDE exploration, yielding a stable and principled learning signal without the $O(G)$ computational burden.

Extensive evaluations across diverse architectures (FLUX.1 and Stable Diffusion) and complex tasks confirm that MPO establishes a new state-of-the-art in both performance and scalability. Most notably, it achieves a 26× wall-clock speedup and a 5× improvement in sample efficiency over prevailing baselines. While MPO currently fixes the initial latent noise during gradient updates to ensure causal disentanglement, future research will explore dynamic noise scheduling to further expand the exploratory manifold. Ultimately, MPO provides a rigorous and highly efficient computational foundation for aligning large-scale generative priors.

6 Acknowledgements

This work was supported in part by the National Natural Science Foundation of China (No.U25A20444, No.62372325, No.62502344), Natural Science Foundation of Tianjin Municipality (No.23JCZDJC00280), , Shandong Province National Talents Supporting Program (No.2023GJJLJRC-070), , Shandong Project towards the Integration of Education and Industry (No.2024ZDZX11)

References

1. Auer, P., Cesa-Bianchi, N., Fischer, P.: Finite-time analysis of the multiarmed bandit problem. Machine learning **47**(2), 235–256 (2002)
2. Bengio, Y., Louradour, J., Collobert, R., Weston, J.: Curriculum learning. In: Proceedings of the 26th annual international conference on machine learning. pp. 41–48 (2009)
3. Betker, J., Goh, G., Jing, L., Brooks, T., Wang, J., Li, L., Ouyang, L., Zhuang, J., Lee, J., Guo, Y., et al.: Improving image generation with better captions. Computer Science. https://cdn. openai. com/papers/dall-e-3. pdf (2023)
4. Black, K., Janner, M., Du, Y., Kostrikov, I., Levine, S.: Training diffusion models with reinforcement learning. arXiv preprint arXiv:2305.13301 (2023)
5. Chen, X., Wu, Z., Liu, X., Pan, Z., Liu, W., Xie, Z., Yu, X., Ruan, C.: Janus-pro: Unified multimodal understanding and generation with data and model scaling. arXiv preprint arXiv:2501.17811 (2025)

6. Dong, H., Xiong, W., Goyal, D., Zhang, Y., Chow, W., Pan, R., Diao, S., Zhang, J., Shum, K., Zhang, T.: Raft: Reward ranked finetuning for generative foundation model alignment. arXiv preprint arXiv:2304.06767 (2023)
7. Esser, P., Kulal, S., Blattmann, A., Entezari, R., Müller, J., Saini, H., Levi, Y., Lorenz, D., Sauer, A., Boesel, F., Podell, D., Dockhorn, T., English, Z., Lacey, K., Goodwin, A., Marek, Y., Rombach, R.: Scaling rectified flow transformers for high-resolution image synthesis (2024), https://arxiv.org/abs/2403.03206
8. Fan, Y., Watkins, O., Du, Y., Liu, H., Ryu, M., Boutilier, C., Abbeel, P., Ghavamzadeh, M., Lee, K., Lee, K.: Dpok: Reinforcement learning for fine-tuning text-to-image diffusion models. Advances in Neural Information Processing Systems **36**, 79858–79885 (2023)
9. Fei, S., Wang, S., Ji, L., Li, A., Zhang, S., Liu, L., Hou, J., Gong, J., Zhao, X., Qiu, X.: Srpo: Self-referential policy optimization for vision-language-action models. arXiv preprint arXiv:2511.15605 (2025)
10. Gao, L., Schulman, J., Hilton, J.: Scaling laws for reward model overoptimization. In: International Conference on Machine Learning. pp. 10835–10866. PMLR (2023)
11. Geist, M., Pietquin, O.: Kalman temporal differences. Journal of artificial intelligence research **39**, 483–532 (2010)
12. Ghosh, D., Hajishirzi, H., Schmidt, L.: Geneval: An object-focused framework for evaluating text-to-image alignment. Advances in Neural Information Processing Systems **36**, 52132–52152 (2023)
13. Gong, L., Hou, X., Li, F., Li, L., Lian, X., Liu, F., Liu, L., Liu, W., Lu, W., Shi, Y., et al.: Seedream 2.0: A native chinese-english bilingual image generation foundation model. arXiv preprint arXiv:2503.07703 (2025)
14. Han, Q., Khamaru, K., Zhang, C.H.: Ucb algorithms for multi-armed bandits: Precise regret and adaptive inference. arXiv preprint arXiv:2412.06126 (2024)
15. Ho, J., Jain, A., Abbeel, P.: Denoising diffusion probabilistic models. Advances in neural information processing systems **33**, 6840–6851 (2020)
16. Hurst, A., Lerer, A., Goucher, A.P., Perelman, A., Ramesh, A., Clark, A., Ostrow, A., Welihinda, A., Hayes, A., Radford, A., et al.: Gpt-4o system card. arXiv preprint arXiv:2410.21276 (2024)
17. Kalman, R.E.: A new approach to linear filtering and prediction problems (1960)
18. Kingma, D.P., Welling, M.: Auto-encoding variational bayes. arXiv preprint arXiv:1312.6114 (2013)
19. Kirstain, Y., Polyak, A., Singer, U., Matiana, S., Penna, J., Levy, O.: Pick-a-pic: An open dataset of user preferences for text-to-image generation. Advances in neural information processing systems **36**, 36652–36663 (2023)
20. Labs, B.F.: Flux. https://github.com/black-forest-labs/flux (2024)
21. Lee, K., Liu, H., Ryu, M., Watkins, O., Du, Y., Boutilier, C., Abbeel, P., Ghavamzadeh, M., Gu, S.S.: Aligning text-to-image models using human feedback. arXiv preprint arXiv:2302.12192 (2023)
22. Li, J., Cui, Y., Huang, T., Ma, Y., Fan, C., Yang, M., Zhong, Z.: Mixgrpo: Unlocking flow-based grpo efficiency with mixed ode-sde. arXiv preprint arXiv:2507.21802 (2025)
23. Lipman, Y., Chen, R.T., Ben-Hamu, H., Nickel, M., Le, M.: Flow matching for generative modeling. arXiv preprint arXiv:2210.02747 (2022)
24. Liu, J., Liu, G., Liang, J., Li, Y., Liu, J., Wang, X., Wan, P., Zhang, D., Ouyang, W.: Flow-grpo: Training flow matching models via online rl. arXiv preprint arXiv:2505.05470 (2025)
25. Loshchilov, I., Hutter, F.: Decoupled weight decay regularization. arXiv preprint arXiv:1711.05101 (2017)

26. Ma, Y., Liu, X., Chen, X., Liu, W., Wu, C., Wu, Z., Pan, Z., Xie, Z., Zhang, H., Yu, X., et al.: Janusflow: Harmonizing autoregression and rectified flow for unified multimodal understanding and generation. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 7739–7751 (2025)
27. Moskovitz, T., Singh, A.K., Strouse, D., Sandholm, T., Salakhutdinov, R., Dragan, A.D., McAleer, S.: Confronting reward model overoptimization with constrained rlhf. arXiv preprint arXiv:2310.04373 (2023)
28. Øksendal, B.: Stochastic differential equations. In: Stochastic differential equations: an introduction with applications, pp. 38–50. Springer (2003)
29. OpenAI: Dalle-2 (2023), `https://openai.com/dall-e-2`
30. Ouyang, L., Wu, J., Jiang, X., Almeida, D., Wainwright, C., Mishkin, P., Zhang, C., Agarwal, S., Slama, K., Ray, A., et al.: Training language models to follow instructions with human feedback. Advances in neural information processing systems **35**, 27730–27744 (2022)
31. Peng, X.B., Kumar, A., Zhang, G., Levine, S.: Advantage-weighted regression: Simple and scalable off-policy reinforcement learning. arXiv preprint arXiv:1910.00177 (2019)
32. Podell, D., English, Z., Lacey, K., Blattmann, A., Dockhorn, T., Müller, J., Penna, J., Rombach, R.: Sdxl: Improving latent diffusion models for high-resolution image synthesis. arXiv preprint arXiv:2307.01952 (2023)
33. Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al.: Learning transferable visual models from natural language supervision. In: International conference on machine learning. pp. 8748–8763. PmLR (2021)
34. Rafailov, R., Sharma, A., Mitchell, E., Manning, C.D., Ermon, S., Finn, C.: Direct preference optimization: Your language model is secretly a reward model. Advances in neural information processing systems **36**, 53728–53741 (2023)
35. Rombach, R., Blattmann, A., Lorenz, D., Esser, P., Ommer, B.: High-resolution image synthesis with latent diffusion models. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 10684–10695 (2022)
36. Saharia, C., Chan, W., Saxena, S., Li, L., Whang, J., Denton, E.L., Ghasemipour, K., Gontijo Lopes, R., Karagol Ayan, B., Salimans, T., et al.: Photorealistic text-to-image diffusion models with deep language understanding. Advances in neural information processing systems **35**, 36479–36494 (2022)
37. Schaul, T., Quan, J., Antonoglou, I., Silver, D.: Prioritized experience replay. arXiv preprint arXiv:1511.05952 (2015)
38. Schuhmann, C., Beaumont, R., Vencu, R., Gordon, C., Wightman, R., Cherti, M., Coombes, T., Katta, A., Mullis, C., Wortsman, M., et al.: Laion-5b: An open large-scale dataset for training next generation image-text models. Advances in neural information processing systems **35**, 25278–25294 (2022)
39. Schulman, J., Wolski, F., Dhariwal, P., Radford, A., Klimov, O.: Proximal policy optimization algorithms. arXiv preprint arXiv:1707.06347 (2017)
40. Song, Y., Sohl-Dickstein, J., Kingma, D.P., Kumar, A., Ermon, S., Poole, B.: Score-based generative modeling through stochastic differential equations. arXiv preprint arXiv:2011.13456 (2020)
41. Wallace, B., Dang, M., Rafailov, R., Zhou, L., Lou, A., Purushwalkam, S., Ermon, S., Xiong, C., Joty, S., Naik, N.: Diffusion model alignment using direct preference optimization. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 8228–8238 (2024)

42. Wang, X., Zhang, X., Luo, Z., Sun, Q., Cui, Y., Wang, J., Zhang, F., Wang, Y., Li, Z., Yu, Q., et al.: Emu3: Next-token prediction is all you need. arXiv preprint arXiv:2409.18869 (2024)
43. Wang, Y., Zang, Y., Li, H., Jin, C., Wang, J.: Unified reward model for multimodal understanding and generation. arXiv preprint arXiv:2503.05236 (2025)
44. Williams, R.J.: Simple statistical gradient-following algorithms for connectionist reinforcement learning. Machine learning **8**(3), 229–256 (1992)
45. Wu, X., Hao, Y., Sun, K., Chen, Y., Zhu, F., Zhao, R., Li, H.: Human preference score v2: A solid benchmark for evaluating human preferences of text-to-image synthesis. arXiv preprint arXiv:2306.09341 (2023)
46. Xie, E., Chen, J., Zhao, Y., Yu, J., Zhu, L., Wu, C., Lin, Y., Zhang, Z., Li, M., Chen, J., et al.: Sana 1.5: Efficient scaling of training-time and inference-time compute in linear diffusion transformer. arXiv preprint arXiv:2501.18427 (2025)
47. Xie, J., Mao, W., Bai, Z., Zhang, D.J., Wang, W., Lin, K.Q., Gu, Y., Chen, Z., Yang, Z., Shou, M.Z.: Show-o: One single transformer to unify multimodal understanding and generation. arXiv preprint arXiv:2408.12528 (2024)
48. Xu, J., Liu, X., Wu, Y., Tong, Y., Li, Q., Ding, M., Tang, J., Dong, Y.: Imagereward: Learning and evaluating human preferences for text-to-image generation. Advances in Neural Information Processing Systems **36**, 15903–15935 (2023)
49. Xue, Z., Wu, J., Gao, Y., Kong, F., Zhu, L., Chen, M., Liu, Z., Liu, W., Guo, Q., Huang, W., et al.: Dancegrpo: Unleashing grpo on visual generation. arXiv preprint arXiv:2505.07818 (2025)
50. You, Z., Cai, X., Gu, J., Xue, T., Dong, C.: Teaching large language models to regress accurate image quality scores using score distribution. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 14483–14494 (2025)
51. Zheng, K., Chen, H., Ye, H., Wang, H., Zhang, Q., Jiang, K., Su, H., Ermon, S., Zhu, J., Liu, M.Y.: Diffusionnft: Online diffusion reinforcement with forward process. arXiv preprint arXiv:2509.16117 (2025)
52. Ziegler, D.M., Stiennon, N., Wu, J., Brown, T.B., Radford, A., Amodei, D., Christiano, P., Irving, G.: Fine-tuning language models from human preferences. arXiv preprint arXiv:1909.08593 (2019)